Feature Selection

What is feature selection?

Feature selection is the process of analysing the explanatory variables in a model, looking for any redundant or irrelevant features, and selecting the subset of predictor variables that improves the prediction and generalisation ability of a model. It is an important step in any predictive modelling task and is used in nearly all predictive models. When building a predictive model, we must think about all the factors a target variable can depend on and the factors which can affect the target variable significantly. Sometimes we may find that there is a large quantity of initial features to be tested, that can include redundant and irrelevant features which we need to identify and remove from the model. The main goal of feature selection is to identify and analyse how the predictors are contributing and their relevance for making predictions on a target variable. The feature selection techniques help identify and select the most significant features to be used for training a machine learning model.

For preparing any dataset for a machine learning task, we need to go through the following key steps: data cleaning, feature selection, data transformation, feature engineering, dimensionality reduction. Although both feature selection and dimensionality reduction aim to reduce the number of input features in the dataset, the main difference between feature selection and dimensionality reduction is that dimensionality reduction projects or transforms the input data to a lower dimensional feature space, whereas the feature selection removes some of the redundant and irrelevant features and only selects a subset of most relevant features.

Why is the feature selection important?

A good selection of features can significantly improve the accuracy and model stability, so that the model will have a good predictive power and generalisation ability (perform well on an unseen data). One important realisation is that although it can seem reasonable to include as many features as possible that can possibly influence the target variable, because including more features can increase the likelihood of having relevant features in the feature set, it is usually not a good practise to include everything in the final model. There are a lot of disadvantages associated with including a lot of features for training a model, such as slow training speed, poor accuracy and complexity. The greater the number of features a model has, the longer is the training time for a machine learning algorithm.

There are two risks that can be associated with including large number of features in a model: Irrelevant features and redundant features.

The irrelevant features are the features that do influence the target variable significantly, and they usually have a weak correlation with the output. Including those features increases the model complexity and creates more uncertainty in the model, reducing the generalisation ability, although it might perform well in the training set. This can lead to overfitting, where a model performs well in the training set but poorly on the unseen data in the test set due of high uncertainly created by the irrelevant features.

The features which are correlated to other features are called redundant features. Redundant features do not add any extra information. There are also ill-conditioning problems associated with having highly correlated features in the training set, as it violates the assumption that the columns of the feature matrix are independent in a regression task, for finding the optimum parameters. In order to achieve a reasonable loss, more iterations would be required in the gradient descent, and therefore the model with correlated features takes longer to train. The presence of redundant features reduces the predictive ability, and the model is difficult to interpret.

The models that are complex and include irrelevant features, usually have a high variance, which means they are sensitive to changes in the training data and are prone to overfitting. On the other hand, the overly simple models, with not enough relevant features, have a high bias and likely to make unreliable predictions. It is important to find the balance between bias and variance by selecting the optimum model complexity. The number of features is one of the main indicators for determining the model complexity.

How can the feature selection help?
The aim of feature selection is to ensure that there are no insignificant features that can hurt the model performance by adding uncertainty, select the most relevant features and detect any redundancy. Feature selection not only helps to reduce the computational cost, as the machine learning algorithm trains faster with fewer number of features, it also prevents overfitting and help improve the performance of the model. The simplified model is easier to interpret and explain. The main benefits of using feature selection include shorter training time, reduced complexity, improved accuracy (if the relevant subset of features is selected), avoiding the curse of dimensionality and reducing the risk of overfitting. In other words, it helps build simpler, faster and more reliable machine learning models.

Feature selection techniques:
There are many different types of feature selection techniques used in practice.
They can be put into two main categories: supervised and unsupervised feature selection. Under the supervised feature selection techniques, there are three further categories: wrapper, filter and embedded methods. There are some libraries and built-in functions in python that can help perform automatic feature selection, or a feature selection code can be written for selecting the features based on certain conditions and requirements.

Looking at the model coefficients is one of the simplest techniques to find the most relevant features. Given that the data was normalised before fitting a model to bring all the variables to a same scale, the coefficients of a models can provide a good guidance towards how a certain feature affects the outcome, depending on the sign and magnitude of the coefficient value. The more significant features tend to have coefficients higher in magnitude. It is important to note that judging the feature importance by coefficients only makes sense when all the features are in the same scale or normalised before performing a machine learning algorithm to find optimum coefficients. This method most closely relates to the intrinsic feature selection method, where some machine learning algorithms, like lasso, automatically shrinks the coefficients of insignificant features to zero, thus removing them from the model.

For features and target variables that are continuous, scatter plots can be used to observe the type of correlation between the features and the target variable, by creating a visual representation of scatter plot matrix, showing the relationship between each feature against the target variable and among themselves. It can indicate whether there is a linear, non-linear or no relation between the variables, and can also guide towards any transformation that could be applied to a feature.

Unsupervised feature selection:
These are the techniques that does not use the target variable for selecting feature importance. It is mainly used to remove redundant and collinear features, which are highly correlated with some other feature, such that one feature can be written as a linear combination of another feature. The redundant variables do not add much additional information to the model. If any

two features are correlated, we can predict one from the other, so the model only needs one of them, as the second feature will not add much extra information.

Some common ways which can be used to select features using unsupervised methods include removing features with a lot of missing values, removing features which have a low variance, and removing highly correlated features.

One can define a threshold for the number of missing values and remove all the features have significantly high missing values and fill all other missing values using some technique, such as the mean or mode values.

The Variance threshold can be used for deciding which features to eliminate based on the variance. One could define a variance threshold and remove all the features which have a variance lower than the variance threshold. The features with a lower variance do not vary much and thus tend to have a low predictive power. Features with higher variance tend to contain more useful information.

One can create a correlation matrix to check the correlation between features and sort the correlation values between features to detect the pair of variables that are highly correlated and remove one of them in order to reduce dimensionality of the feature set without much loss of information, keeping the feature which has a stronger relation with the target variable. The Pearson's correlation is commonly used to check for linear relationship and correlation between features. The Spearman's rank correlation and Kendall's rank correlation can be used to check for a non-linear relationship as well. Variance inflation factor (VIF) and the condition number are also commonly used methods to detect multicollinearity. Given a covariance matrix, the condition number is defined as the largest eigenvalue/ the smallest eigenvalue. A high value of condition number (>30) indicates a strong multicollinearity. A condition number also measures how sensitive the model is to small perturbations in the input data. A variance inflation factor is used in ordinary least square regression analysis to check for multicollinearity. It measures the increase in variance when the predictor is included in the model with other predictors, and a large value indicates a highly collinear relationship with other predictors.

Supervised feature selection:

These are the techniques that use the target variable for selecting feature importance. It is mainly used to remove irrelevant features. The features are selected based on their relationship with the target, found using statistical methods, feature importance methods and assessing the model performance with a subset of features. To detect the irrelevant features, we test whether the outcome is significantly affected by a given feature.

Under the supervised feature selection, there are three main types of techniques commonly used: filter methods, wrapper methods and embedded/intrinsic methods.

Filter methods:

The filter methods are used as a data pre-processing step, where the selection of features is independent of any machine learning algorithm used. The features ranked and selected based on some statistical scores, for example the feature's correlation with the response.

The filter-based feature selection methods commonly use the univariate statistical measures for examining the feature importance instead of the model testing using cross-validation. Each predictor is evaluated separately, using the statistical tests to examine the strength of relationship between a feature variable and response. It does not use a machine learning method to assess the features, and is therefore much faster than the wrapper methods, as it does not involve training the models which can be time consuming. This method is very useful when working with a large feature set or a high dimensional data, as training various models to assess feature importance could be very computationally expensive. The limitation of using a filter method is that it can fail to find the best subset of features when there is not enough data to model the statistical correlation of features with the response. Another limitation is that the interaction between input variables is not considered, which means that the important but redundant variables might be selected, thus we need to take care of the collinearity problem as well. The features should be correlated with the target variable but uncorrelated among themselves. Using the filtering methods in the pre-processing steps, to select the most relevant features for model training, reduces the risk of overfitting the model and hence improve the generalisation ability, giving a better accuracy on the test set or unseen data.

The kind of univariate statistic used to rank features depend on the type of data, and whether a feature and target variable is continuous or categorical.

Categorical feature-categorical response: Chi-squared test, Mutual information
Continuous feature-categorical response: Anova, Kendall's rank coefficient
Categorical feature-continuous response: Anova, Kendall's rank coefficient
Continuous feature-continuous response: Pearson correlation, Spearman's correlation, Mutual information

Sklearn.feature_selection module is very useful for the feature selection tasks. Some useful statistical measures or scoring functions provided by the Scikit-learn library are: f_regression (Pearson's correlation coefficient for continuous feature-continuous response), mutual_info_regression (mutual information for continuous feature-continuous response), f_classif (anova test for categorical feature-continuous response and continuous feature-categorical response), chi2 (chi-squared test for categorical feature-categorical response), mutual_info_classif (mutual information for categorical feature-categorical response).
Some popular filtering methods based on some univariate statistics are also provided by the Scikit-learn library such as: SelectKBest (Select the top k variables based on univariate statistical test), RFE (recursive feature elimination based on a particular model, univariate statistic and the number of features to be selected), feature selection using SelectFromModel. The SciPy library provides some more statistical measures such Kendall's tau: kendalltau, and Spearman's rank correlation: spearmanr.

Some of the statistical methods used to understand the importance of features/ evaluate feature relevance (scoring methods):

Kendall's rank coefficient- It is used for ordinal categorical features, can be used to capture non-linear relationship.

Chi squared test- It can be used to test for any dependence between independently sampled categorical features, with expected frequency of more than 5. The null hypothesis assumes that there is no relationship between a feature and target, and if a feature does affect the target, then we expect a low p value. It can be used to detect any type of relation between feature and response, including non-monotonic.

Fisher's score- It uses the concept that the distances between samples in different classes should be as high as possible and distance between samples in the same class should be as small as possible.

Variance threshold/ low variance filter- It is used to select features based on their variance. It assumes that the features with high variance may contain more useful information. It is an unsupervised feature selection technique.

Dispersion ratio- It is defined as the ratio between the arithmetic mean and geometric mean. A higher value of the ratio implies a higher dispersion and a more relevant feature.

Permutation feature importance: It is defined to be the decrease in model score when a single feature is randomly shuffles. The more the model depends on that feature, the higher will be the drop in the model score. It can detect a non-linear relation.

Pearson's correlation- It is used for continuous features and continuous target variable. It measures the strength of linear relationship between two variables. It gives a low value for a non-linear relation, so does not capture strong non-linear relationships between two variables.

Spearman's correlation- It does not make any assumption about the distribution and can be used even when the variable set is not normally distributed. It can be used to detect linear or non-linear relationship from any distribution. Spearman's correlation tests if a monotonic relationship exists but performs poorly when the relationship is non-monotonic. Chi-squared test can be used to identify a non-monotonic relation.

Maximal information coefficient- It measure of strength of linear or non-linear relationship between two variables.

Distance correlation- It is a measure of dependence and strength of linear or non-linear relation between two random variables.

Analysis of variance (anova)- It is used for normally distributed data. It compares the mean of different groups and can be used to test whether the differences between groups/variables are statistically significant, and if a target variable is significantly impacted by a feature. It can detect if a feature is independent of the target variable and thus irrelevant. If a feature is relevant, we expect to see significant differences between the variances.

Mutual information- It is a quantity that measures how knowing the information about one random variable affects the other random variable. It can capture any kind of statistical dependency.

Wrapper methods:
The wrapper methods use a machine learning algorithm to test on different subsets of input features and uses its performance for evaluation in order to judge the feature importance. The wrapper method feature selection is based on a particular machine learning algorithm, where a different combination of features is used to train a model, and their performance is evaluated and compared to other combinations, using a same predictive model, and performance scores are assigned. There are two hyperparameters that can be involved in the wrapper methods for feature selection: the number of features to select and the algorithm used to help choose features.

Some common ways a wrapper feature selection method can be applied include- using forward feature selection, using backward feature elimination, using a stepwise selection/hybrid method, using exhaustive feature selection method, and a recursive feature elimination (RFE).

Forward feature selection:
It is an iterative method, where we start with having no features in the model, and at each iteration, a new feature is added which best improves the model performance. A model is run for each individual feature in the first step, and the one giving a best score is selected. In the next step, a second feature is added with the first feature already selected in the previous step, and the model is trained again with all the remaining features, to select the best second feature based on some model performance score. This process is repeated until an addition of a new feature does not significantly improve the model performance. The features that were not selected in the process are excluded from the model and the selected ones are used for training.

Backward feature elimination:
It is an iterative method, where we start with all the features in the model, and at each iteration, the least significant feature is removed which results in the best improvement in model performance. This process is repeated until no improvement is observed on removing any feature. Only the features that remain in the model are used for training.

Stepwise selection/Hybrid method:
It is a combination of forward selection and backward elimination. It starts with zero features and adds the first two features like in the forward selection, and then from the next step onwards, it adds another feature but also removes a feature if it seems insignificant after training the model with new set of features.
A hybrid method combines the filter and wrapper technique to generate a feature ranking and then use the top k features from this list to perform wrapper methods. The reduced feature space is used to improve the computational time of the wrapper methods.

Some drawbacks for forward selection, backward elimination and stepwise selection methods include that they do not run through all possible combinations of features and the search is not exhaustive, although these methods are more feasible and computationally efficient in practice. A model could also have a high multicollinearity, as these methods do not test for the relationship among the features.

Exhaustive feature selection:
A greedy search approach could be applied to search for the space of all possible subsets of features and evaluating all possible combinations using a machine learning algorithm, against some evaluation criteria. Due to its exhaustive search, training machine learning model with each feature subset, this method can be used to provide a best subset of features, based on a particular evaluation criterion. It usually results in better predictive accuracy than filter methods. However, it is not the best method for very large datasets due to being computationally costly.

Recursive feature elimination (RFE):
It is a greedy search optimisation algorithm which aims to find the best performing subsets of features, given how many features are preferred. It is a common method that is used for selecting the best certain number of features for training a model. Starting with all features in the model, a given machine learning algorithm is fitted and used to rank the features by importance using a score that is either model based or using a statistical method. The least

important feature is removed at every step until a specified number of features remain in the model. It is possible to automatically select the optimum number of features before applying the RFE.

Embedded/intrinsic methods:
These are the algorithms that perform automatic feature selection during the training process. It usually includes the regularisation methods such as lasso, elastic net, ridge regression, as well as decision trees and random forest. Regularisation methods penalise a feature given a threshold for the sum of coefficients, introducing additional constraints in the optimisation task, such that the model is biased towards lower complexity by shrinking the coefficients of insignificant features towards zero. It is important to scale all the feature variables before applying a regularisation method.

How to train and test data with feature selection:
For any predictive modelling task, the data is generally split into a training set, validation set, and a testing set. A large part of the data is used for model training from which the model is trained, a part of data is used for hyperparameter tuning, which helps in selecting the hyperparameters for the model to use in model training, and another part of data is used for the model evaluation to test how the model performs on an unseen data.
It is important to ensure that only the training data/ training folds are being used to select predictors in cross validation. The feature selection step should be included before using the data for model training. When using cross-validation, feature selection should be performed on the data fold right before the model is trained. Feature selection should not be performed first to prepare the data, followed by model selection and training and testing on the selected features. If the feature selection is performed on the whole dataset, followed by cross-validation, this implies that the test data in each fold was also used to choose the features, resulting in a biased model performance. The test set should be independent of the training set and not be used in any data preparation tasks such as filling missing values, normalising and feature selection, as it creates a data leakage and biased results.
Finally, we need to make sure that the training accuracy is not significantly higher than the test accuracy as this can indicate an overfitting. A feature selection technique can be used to deal with the problem of overfitting by selecting the most significant features and simplifying the model.