FOUNDATIONS OF MACHINE LEARNING AND DEEP LEARNING

FEATURE SELECTION

Group Members

Abigail Naa Amankwaa Abeo aabeo@aimsammi.org
Etienne Ntumba Kabongo enkabongo@aimsammi.org
Jean Robin Raherisambatra jrraherisambatra@aimsammi.org
Souleymane Balde sbalde@aimsammi.org

African Masters in Machine Intelligence

Supervised by
Moustapha CISSE

April 27, 2023

Abstract

Feature selection is an essential technique in the field of machine learning that aims to identify and select the most relevant features from a larger set of potential features. The primary goal of feature selection is to improve the accuracy, efficiency, and interpretability of machine learning models by reducing the dimensionality of input data and eliminating irrelevant or redundant features. There are various techniques of feature selection available, including filter methods, wrapper methods, and embedded methods. The selection of the most suitable method depends on the specific needs of the problem, such as the size and quality of the dataset, the complexity of the model, and the interpretability of the features. In this project, we provide an overview of the different types of feature selection methods. Overall, feature selection is a critical step in building accurate and interpretable machine learning models, and its importance is likely to grow as more complex and high-dimensional datasets become prevalent in various domains.

Table of Contents

List of Figures

1. Feature Selection

Machine learning or deep learning models have become essential tools in solving many problems today. However, for their construction, many things have to be taken into account, in particular the right features to use in your data. It is almost rare that all the variables in the dataset are useful for building a model. Adding redundant variables reduces the model's generalization capability and may also reduce the overall accuracy. Also, adding more variables to a model increases the overall complexity of the model.

In this work, we will look at what feature selection is, the different methods of feature selection and how they work.

Feature selection involves reducing the number of input features to your model by using only relevant features and getting rid of the irrelevant ones in the data [1]. It is the process of choosing relevant features for your machine learning model based on the type of problem you are trying to solve. We do this by including or excluding important or unimportant features respectively without changing them. It helps in cutting down the noise in our data and reducing the size of our input data.

Figure 1.1: An Illustration of Feature Selection

1.1 Why Feature Selection?

There are quite a few reasons why feature selection is important. Some of these reasons are:

1.1.1 Irrelevant and redundant features

Some features might be irrelevant to the problem at hand. This means they have no relation with the target variable and are completely unrelated to the task the model is designed to solve.

Discarding irrelevant features will prevent the model from picking up spurious correlations, thus fending off overfitting.

1.1.2 Curse of dimensionality

Feature selection techniques are especially indispensable in scenarios with many features but few training examples. Such cases suffer from what is known as the curse of dimensionality. The solution is to decrease the dimensionality of the features space, for instance, via feature selection.

1.1.3 Training time

The more features we have, the more training time we get. The specifics of this trade-off depend on the particular learning algorithm being used, but in situations where retraining needs to happen in real-time, one might need to limit oneself to a couple of best features.

1.1.4 Interpretability

With too many features, we lose the explainability of the model. While not always the primary modeling goal, interpreting and explaining the model's results are often important and, in some regulated domains, might even constitute a legal requirement.

1.1.5 Occam's Razor

According to this law of parsimony, simpler models should be preferred over the more complex ones as long as their performance is the same. This also has to do with the machine learning engineer's nemesis, overfitting. Less complex models are less likely to overfit the data.

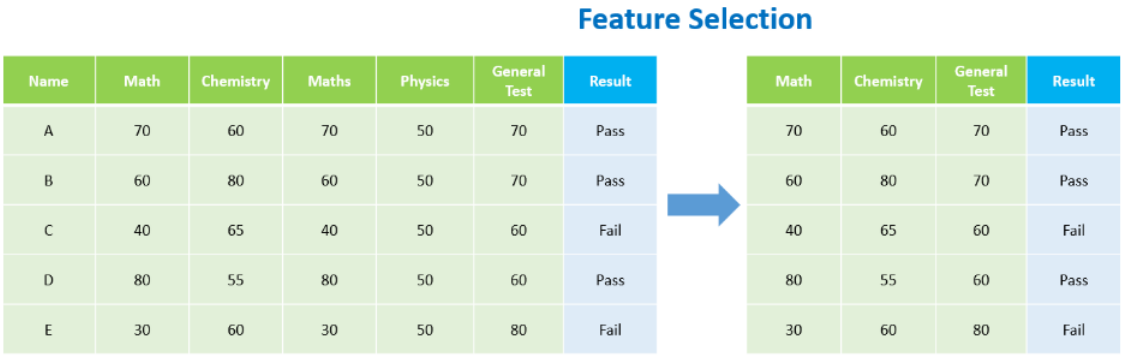

Figure 1.2: Example of feature selection

1.2 Filter Method

Filter method feature selection is a popular technique used in machine learning and data analysis to identify the most relevant features (also called variables or predictors) that can best explain

the outcome variable. The filter method involves evaluating the intrinsic characteristics of each feature and ranking them based on their relevance to the target variable, without considering the interaction between features [2].

The main idea behind the filter method is to use statistical metrics, such as correlation, mutual information, chi-squared, and ANOVA, to measure the strength of the relationship between each feature and the target variable. The higher the metric value, the more important the feature is considered to be[3].

Once the features are ranked based on the metric value, a threshold is set to determine which features to keep and which ones to discard. The most common approach is to keep the top-k features, where k is a predefined number or a percentage of the total number of features[4]. Alternatively, a cutoff value can be set for the metric score, and any feature with a score below that value is removed [4].

Filter method feature selection is simple and fast compared to other feature selection methods. It can handle a large number of features and can be applied to various types of datasets. However, it has some limitations, such as ignoring the interaction between features and assuming that all features are independent. Therefore, it is recommended to use filter method feature selection in combination with other feature selection methods or data preprocessing techniques to improve the model's performance[5].

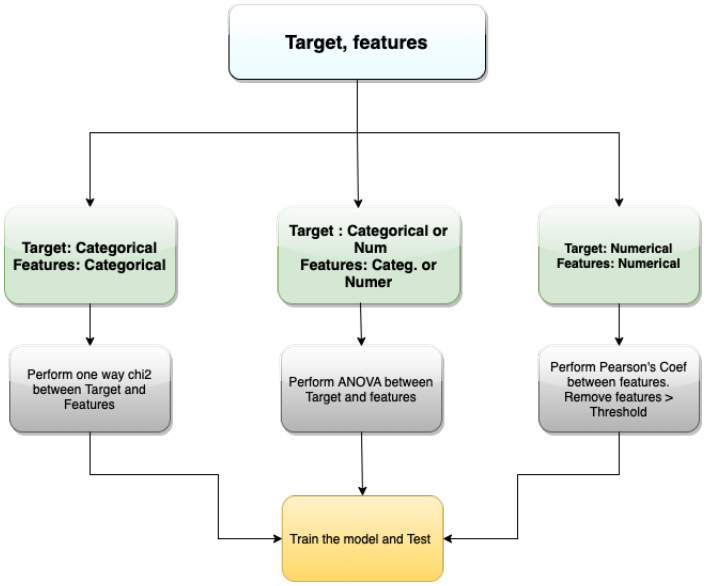

Figure 1.3: Filter Method Summary

1.2.1 Pearson's Correlation

Pearson's correlation coefficient is a measure of the linear relationship between two variables, and it is used for filter feature selection[6]. To apply Pearson's correlation coefficient for feature selection, the following steps are taken:

- Compute the Pearson's correlation coefficient between each feature and the target variable.
- Rank the features based on their absolute value of the correlation coefficient in descending order.
- Select the top k features with the highest absolute correlation coefficients.

The idea behind this approach is to select features that have the strongest linear relationship with the target variable. However, it is important to note that Pearson's correlation coefficient only captures linear relationships, and may not be suitable for non-linear relationships [7].

It is also important to be cautious when selecting features based on correlation, as highly correlated features may introduce multicollinearity and negatively impact model performance. In such cases, techniques such as regularization or principal component analysis (PCA) can be used to address multicollinearity [8].

It is used as a measure for quantifying linear dependence between two continuous variables X and Y. Its value varies from -1 to $+1$. Pearson's correlation is given as:

$$\rho_{X,Y} = \frac{cov(X,Y)}{\delta_X \delta_Y}$$

1.2.2 ANOVA F-test for Feature Selection

ANOVA (Analysis of Variance) F-test is a statistical test that is used for filter feature selection in machine learning. ANOVA F-test is used to determine whether the means of two or more groups are significantly different from each other, based on the variance between the groups and the variance within the groups [9].

To apply ANOVA F-test for feature selection, the following steps are taken:

- Group the data based on the target variable.
- Compute the mean and variance of each feature for each group.
- Compute the ANOVA F-value for each feature, which is the ratio of the variance between the groups to the variance within the groups.
- Rank the features based on their ANOVA F-values in descending order.
- Select the top k features with the highest ANOVA F-values.

The idea behind this approach is to select features that have the largest variance between the groups and the smallest variance within the groups. This indicates that the feature has a significant effect on the target variable and can be a good predictor [10].

It is important to note that ANOVA F-test assumes that the data is normally distributed and the variances are equal between the groups. Additionally, like Pearson's correlation coefficient, ANOVA F-test only captures linear relationships, and may not be suitable for non-linear relationships[11]. Also, ANOVA stands for Analysis of variance. It is similar to LDA except for the fact that it is operated using one or more categorical independent features and one continuous dependent feature. It provides a statistical test of whether the means of several groups are equal or not [12].

1.2.3 Chi-Square

Chi-square test is a statistical test that is used for filter feature selection in machine learning. This statistical test is applied to the groups of categorical features to evaluate the likelihood of correlation or association between them using their frequency distributions. The chi-square test measures the dependence between two categorical variables, and it is commonly used to test the independence of two variables[13].

To apply chi-square test for feature selection, the following steps are taken:

- Construct a contingency table that shows the frequency of occurrence of each feature and the target variable.
- Compute the chi-square statistic for each feature, which is the sum of the squared differences between the observed and expected frequencies of each cell in the contingency table.
- Compute the degrees of freedom for the chi-square test, which is the product of the number of categories in the feature and the number of categories in the target variable minus one.
- Compute the p-value for each feature, which is the probability of observing a chi-square statistic as extreme as or more extreme than the computed value under the null hypothesis of independence.
- Rank the features based on their p-values in ascending order.
- Select the top k features with the smallest p-values.

The idea behind this approach is to select features that are most dependent on the target variable, as indicated by the low p-values. The chi-square test is suitable for categorical data, but it assumes that the expected frequencies are greater than or equal to five in each cell of the contingency table [14].

It is important to note that the chi-square test only captures linear relationships between categorical variables, and may not be suitable for non-linear relationships.

1.3 Wrapper Method

The wrapper methods create several models having different subsets of input feature variables. Then, the subset of features which result in the best performing model in accordance with the performance metric are selected[15].

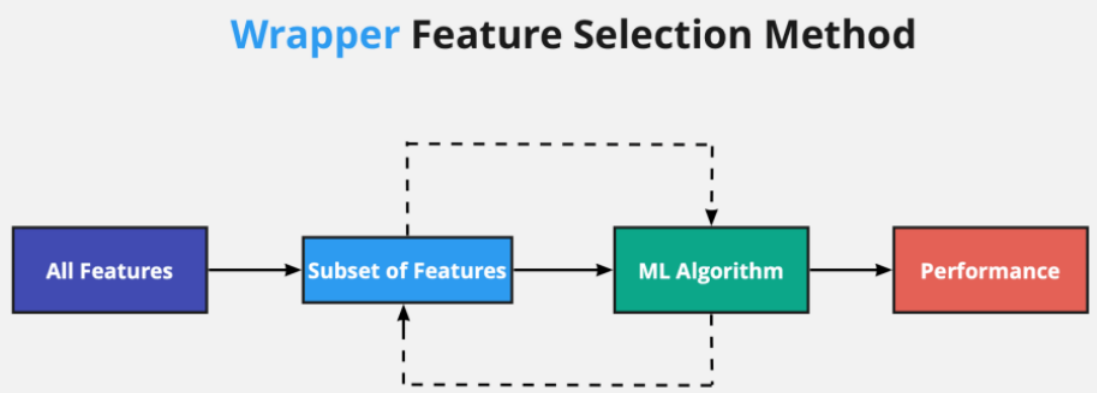

Figure 1.4: Wrapper Method Summary

1.3.1 Forward feature selection

In forward feature selection, the algorithm starts with using just one of the features and tries to model the data using the given model. It then picks the feature that provides the highest accuracy, or a set performance metric. This process repeats itself up to a set number of features that the user decides [16].

Below are the steps of the forward feature selection:

1. Train n models using each feature(n) individually and check the performance.
2. Choose the variable that gives the best performance.
3. Repeat the process and add one variable at a time.
4. Variable producing the highest improvement is retained.
5. Repeat the entire process until the performance on the model drops below a certain threshold or desired number of features is reached.

1.3.2 Backward feature selection

Backward feature selection is an example of wrapper method that selects the most relevant features in a machine learning model by starting with all the features and removing iteratively the least relevant features until a desired level of performance is achieved [16].

Below are the step of the backward feature selection:

1. Start with a model that includes all the features.
2. Train the model and evaluate its performance by using a chosen metric.
3. Remove one feature from the feature set; the one that has least impact on the performance.
4. Train again the model using the remaining features and evaluate its performance.
5. If the performance has not decreased significantly, continue with step 3 and remove another least relevant feature;
6. Repeat step 3 to step 5 until the performance on the model drops below a certain threshold or desired number of features is reached.

1.4 L1 Regularization

Regularization, as the name suggests, consists of settling something. In machine learning, we do this by adding information to the objective function to prevent overfitting. We use regularization because we want to add bias to our model to prevent it from fitting our training data too well. After adding regularization, we end up with a machine learning model that performs well on the training data, and has a good ability to generalize to new examples that it has not seen during training.

1.4.1 Definition

As said before, regularization consists of adding information to the objective function. So for the L1 regularization, we add the L1 norm multiplied by a constant to the objective function.

$$L_1 Regularization = LossFunction + \lambda \|W\|_1$$

Where λ is a constant and W is a weight parameter.

L1 regularization is a form of feature selection, because when we assign a feature with a 0 weight, we are multiplying the feature values by 0 which returns 0, eradicating the significance of that feature. If the input features of our model have weights closer to 0, our data would be sparse. A selection of the input features would have weights equal to zero, and the rest would be non-zero.

1.4.2 Sparsity

In machine learning, sparsity refers to a matrix of numbers that includes many zeros.
When we do an L1 regularization, we force the coefficients to tend towards zero to obtain the maximum of zeros according to the characteristics. So we finally have a sparse matrix, hence the sparsity with the L1 regularization.

Users					Movies						Target
A	B	C	D	E	Parasite	Joker	Avengers	Spotlight	The Great Beauty	There will be blood	Rating
1	0	0	0	0	1	0	0	0	0	0	5
1	0	0	0	0	0	1	0	0	0	0	4
1	0	0	0	0	0	0	1	0	0	0	4
0	1	0	0	0	1	0	0	0	1	0	2
0	1	0	0	0	0	0	0	1	0	0	4
0	1	0	0	0	0	0	0	0	1	0	3
0	0	1	0	0	0	0	1	0	0	0	5
0	0	0	1	0	0	0	0	0	0	1	4
0	0	0	0	1	0	0	1	0	0	0	4

Figure 1.5: Example of Sparse Matrix

1.5 Python Code

The code aims to showcase the filter, wrapper and L1 regularization feature selection methods. Click here to visit the notebook.

1.6 Conclusion

In conclusion, feature selection is an important step in the machine learning pipeline that involves identifying the most relevant features from a larger set of potential features. The goal of feature selection is to improve the accuracy, speed, and interpretability of machine learning models by reducing the dimensionality of the input data and eliminating irrelevant or redundant features.

There are several methods of feature selection, including filter methods, wrapper methods, and embedded methods. Filter methods are based on statistical tests or other measures of feature relevance and can be applied independently of the machine learning model. Wrapper methods involve training and evaluating the machine learning model on subsets of features to identify the most relevant features. Embedded methods incorporate feature selection into the model training process itself, optimizing both feature selection and model performance simultaneously.

Forward feature selection is a popular technique in feature selection that involves iteratively adding the most relevant features to the model until a stopping criterion is met. This approach can be effective, particularly for datasets with a large number of features or limited sample sizes.

Overall, feature selection is a crucial step in developing accurate and effective machine learning

models. By identifying the most relevant features and eliminating irrelevant or redundant ones, feature selection can improve the interpretability, scalability, and generalization performance of machine learning models across a wide range of applications.

1. References

[1] Nazish Naheed, Muhammad Shaheen, Sajid Ali Khan, Mohammed Alawairdhi, and Muhammad Attique Khan. Importance of features selection, attributes selection, challenges and future directions for medical imaging data: a review. *Computer Modeling in Engineering & Sciences*, 125(1):314–344, 2020.

[2] Noelia Sánchez-Maroño, Amparo Alonso-Betanzos, and María Tombilla-Sanromán. Filter methods for feature selection–a comparative study. *Lecture notes in computer science*, 4881:178–187, 2007.

[3] Marianne Cherrington, Fadi Thabtah, Joan Lu, and Qiang Xu. Feature selection: filter methods performance challenges. In *2019 International Conference on Computer and Information Sciences (ICCIS)*, pages 1–4. IEEE, 2019.

[4] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28, 2014.

[5] Alan Jović, Karla Brkić, and Nikola Bogunović. A review of feature selection methods with applications. In *2015 38th international convention on information and communication technology, electronics and microelectronics (MIPRO)*, pages 1200–1205. Ieee, 2015.

[6] Jacek Biesiada and Wlodzisław Duch. Feature selection for high-dimensional data—a pearson redundancy based filter. In *Computer recognition systems 2*, pages 242–249. Springer, 2007.

[7] Matthew Shardlow. An analysis of feature selection techniques. *The University of Manchester*, 1(2016):1–7, 2016.

[8] Inzamam Mashood Nasir, Muhammad Attique Khan, Mussarat Yasmin, Jamal Hussain Shah, Marcin Gabryel, Rafał Scherer, and Robertas Damaševičius. Pearson correlation-based feature selection for document classification using balanced training. *Sensors*, 20(23):6793, 2020.

[9] Mukesh Kumar, Nitish Kumar Rath, Amitav Swain, and Santanu Kumar Rath. Feature selection and classification of microarray data using mapreduce based anova and k-nearest neighbor. *Procedia Computer Science*, 54:301–310, 2015.

[10] Nadir Omer Fadl Elssied, Othman Ibrahim, and Ahmed Hamza Osman. A novel feature selection based on one-way anova f-test for e-mail spam classification. *Research Journal of Applied Sciences, Engineering and Technology*, 7(3):625–638, 2014.

[11] Shaikh Shakeela, N Sai Shankar, P Mohan Reddy, T Kavya Tulasi, and M Mahesh Koneru. Optimal ensemble learning based on distinctive feature selection by univariate anova-f statistics for ids. *International Journal of Electronics and Telecommunications*, pages 267–275, 2021.

[12] Huanjing Wang, Taghi M Khoshgoftaar, and Amri Napolitano. A comparative study of ensemble feature selection techniques for software defect prediction. In *2010 Ninth International Conference on Machine Learning and Applications*, pages 135–140. IEEE, 2010.

[13] Yujia Zhai, Wei Song, Xianjun Liu, Lizhen Liu, and Xinlei Zhao. A chi-square statistics based feature selection method in text classification. In *2018 IEEE 9th International conference on software engineering and service science (ICSESS)*, pages 160–163. IEEE, 2018.

[14] Nachirat Rachburee and Wattana Punlumjeak. A comparison of feature selection approach between greedy, ig-ratio, chi-square, and mrmr in educational mining. In *2015 7th international conference on information technology and electrical engineering (ICITEE)*, pages 420–424. IEEE, 2015.

[15] Golnaz Sahebi, Parisa Movahedi, Masoumeh Ebrahimi, Tapio Pahikkala, Juha Plosila, and Hannu Tenhunen. Gefes: A generalized wrapper feature selection approach for optimizing classification performance. *Computers in biology and medicine*, 125:103974, 2020.

[16] Sahameh Shafiee, Lars Martin Lied, Ingunn Burud, Jon Arne Dieseth, Muath Alsheikh, and Morten Lillemo. Sequential forward selection and support vector regression in comparison to lasso regression for spring wheat yield prediction based on uav imagery. *Computers and Electronics in Agriculture*, 183:106036, 2021.