Contents

1 Introduction

1.1 Definition of Machine Learning

Machine Learning (ML) is a branch of artificial intelligence focused on developing algorithms that allow computers to learn from data and experiences without explicit programming. The goal is to create mathematical models capable of analyzing and interpreting complex patterns and relationships within data, which can then be used to make predictions or inform decisions.
[Academy(2024)]

1.2 Main types of Machine Learning

Machine Learning can be broadly categorized into three main types:

- **Supervised Learning:** The model is trained on labeled data, meaning each input is associated with a corresponding output. Supervised learning is commonly used for tasks such as classification and regression
- **Unsupervised Learning:** The model analyzes unlabeled data, seeking to discover hidden patterns or structures. Typical applications include clustering and dimensionality reduction.
- **Reinforcement Learning:** In this approach, an agent learns by interacting with its environment, receiving rewards or penalties based on its actions, which allows it to optimize its strategy.

[Combeenut.pl(2023)]

2 Basic Concepts in Machine Learning

2.1 Data Splits

To build robust Machine Learning models, data is typically divided into three main subsets:

- **Training Set:** The model is trained on this portion of data to learn the relationships between input features and target labels.
- **Validation Set:** This subset is used to fine-tune model parameters and evaluate the model's performance during training, helping to prevent overfitting.
- **Test Set:** After training and tuning, the test set evaluates the model's performance on unseen data, providing an unbiased assessment of its generalization ability.

This division allows models to be evaluated accurately, ensuring they perform well not only on the data they were trained on but also on new, unseen data.
[GeeksforGeeks(2023a)]

2.2 Features and Labels

In supervised learning, features and labels are fundamental components of a dataset:

- **Features:** Also known as input variables or attributes, features represent the characteristics or properties of the data that help the model make predictions. For example, in a dataset predicting house prices, features might include the square footage, number of rooms, and location.
- **Labels:** Also called target variables, labels are the outputs or results that the model aims to predict. In the same house pricing example, the label would be the actual price of the house.

Together, features and labels allow supervised learning algorithms to learn mappings from inputs to outputs.
[GeeksforGeeks(2023b)]

3 Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) is a crucial step in the machine learning workflow, enabling data scientists to understand the underlying patterns, detect anomalies, and extract meaningful insights from datasets.

3.1 The Role of EDA in Machine Learning

EDA serves several essential functions in the machine-learning process:

- **Understanding Data Distribution:** By analyzing the distribution of variables, EDA helps identify normal distributions and detect outliers.
- **Assessing Data Quality:** EDA aids in identifying missing values, inconsistencies, and errors within the dataset, which are critical to address before model training.
- **Identifying Relationships Between Variables:** Through correlation analysis and pattern recognition, EDA reveals dependencies and interactions among variables, informing feature selection and engineering.
- **Guiding Model Selection:** Insights gained from EDA inform the choice of appropriate algorithms and modeling techniques, enhancing model performance and interpretability.

[MachineLearningPlus(2023), Microsoft(2023), Talent500(2023)]

3.2 Basic Visualization and Insights

Visualization is a fundamental component of EDA, providing intuitive and accessible representations of data. Common visualization techniques include:

- **Histograms and Box Plots:** These plots display the distribution of numerical variables and help identify outliers.
- **Scatter Plots:** Used to examine relationships between two continuous variables, revealing potential correlations or trends.
- **Correlation Matrices:** Heatmaps illustrating the correlation coefficients between variables, aiding in the detection of multicollinearity.
- **Bar Charts and Pie Charts:** Effective for visualizing the distribution of categorical variables and comparing group frequencies.

[MachineLearningPlus(2023), Codecademy(2023)]

4 Classification Models

Classification models are essential algorithms in machine learning that enable prediction of data belonging to specific categories. This section covers the most popular classification models, their key features, advantages, and limitations.

4.1 Logistic Regression

Logistic regression despite its name, is a classification model primarily used for binary classification. It applies the logistic function to predict class probabilities, making it ideal for scenarios like predicting "yes" or "no" outcomes. Logistic regression is easy to interpret and works well when there is a clear linear separation between classes.

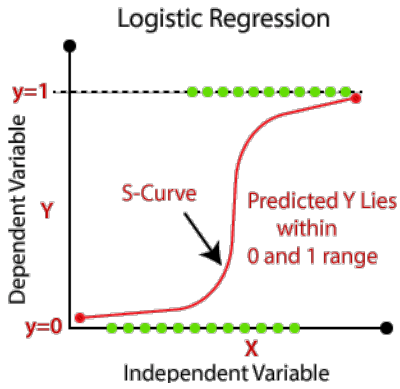

Figure 1: Logistic Regression

Advantages of Logistic Regression:

- **Interpretability:** Logistic regression is relatively easy to understand and interpret, making it suitable for applications where the transparency of results is important.
- **Low Computational Requirements:** This model is less complex compared to more advanced techniques, resulting in shorter training times.
- **Good Performance with Linear Separation:** Logistic regression performs effectively when the data has a linear separation, making it ideal for many real-world applications.

Disadvantages of Logistic Regression:

- **Limitations with Non-Linearity:** Logistic regression does not handle non-linear relationships between variables well, which can limit its effectiveness in more complex problems.
- **Requirement for Linear Separation:** For the model to be effective, the data must be well-separated linearly, which is not always the case in real-world applications.
- **Sensitivity to Outliers:** The model is sensitive to outliers, which can affect its performance and accuracy in predictions.

[learn Developers(2023f), Neverthoughtaboutthat(2023)]

4.2 Support Vector Machine (SVM)

The goal of SVM is to find a hyperplane in the feature space that maximally separates data points from different classes, increasing classification accuracy. The optimal hyperplane maximizes the margin—the distance between the closest data points from each class.

The hyperplane's dimension depends on the number of features: with two features, it's a line; with three, it's a plane. For higher dimensions, kernel functions are used to map data into a higher-dimensional space, making classes more separable.

Support vectors are the data points closest to the hyperplane, which significantly influence its position and allow the classifier's margin to be maximized.

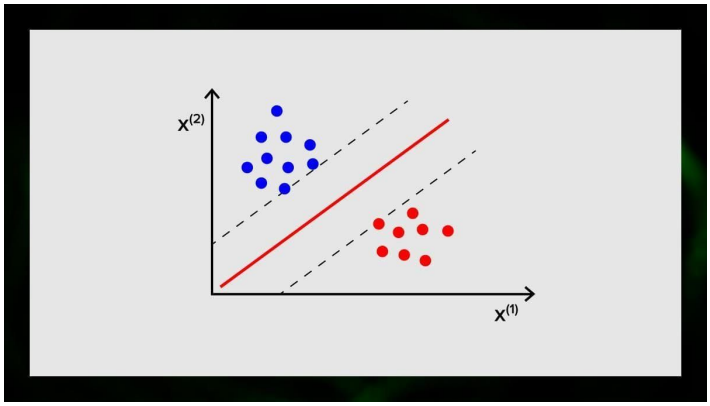

Figure 2: SVM - hyperplane in the feature space

The dimension of the hyperplane depends on the number of features. If the number of input features is 2, the hyperplane is simply a line. If the number of features is 3, the hyperplane becomes a two-dimensional plane.

Advantages of SVM:

- **Effective in High Dimensions:** SVM performs well in high-dimensional spaces and is effective when the number of dimensions exceeds the number of samples.
- **Robust to Overfitting:** Especially in high-dimensional space, SVM is less prone to overfitting compared to other algorithms, particularly when using a proper kernel.
- **Versatile Kernel Trick:** SVM uses kernel functions to handle non-linear classification problems, enabling it to create complex decision boundaries.

Disadvantages of SVM:

- **Choice of Kernel:** The performance of SVM depends on the choice of the kernel. Selecting an inappropriate kernel can lead to poor model performance.
- **Computationally Intensive:** Training SVMs can be computationally intensive, particularly with large datasets, which may lead to longer training times.
- **Less Effective with Noisy Data** SVMs may not perform well with overlapping classes or noisy data, as they aim to maximize the margin without accounting for misclassifications effectively.

[Neverthoughtaboutthat(2023), learn Developers(2023l)]

4.3 Decision Trees

Decision trees are a popular and intuitive class of models used for both classification and regression tasks. They are structured like a tree, with nodes representing decisions based on the values of input features, leading to branches that further split the data until a final decision (or output) is reached at the leaves of the tree.

Structure of Decision Trees:

- Each internal node in the tree represents a feature or attribute of the data, while each branch corresponds to a decision rule
- The leaves of the tree represent the final outcomes or predictions for the input data.
- The path from the root of the tree to a leaf node represents a sequence of decisions that leads to a particular classification or regression outcome.

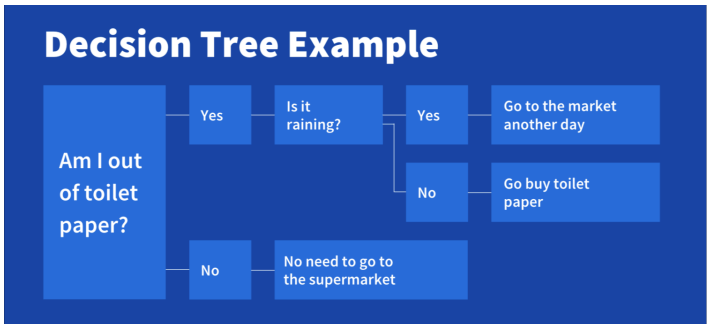

Figure 3: Decision Tree Example

Advantages of Decision Trees:

- **Interpretability:** Decision trees are straightforward to visualize and interpret. Stakeholders can easily understand the model's logic, making them ideal for applications where transparency is important.
- **Non-Linear Relationships:** They can capture non-linear relationships between features without requiring transformation or scaling of the data.
- **No Need for Feature Scaling:** Unlike many other machine learning algorithms, decision trees do not require normalized or standardized data.

Disadvantages of Decision Trees:

- **Overfitting:** One of the most significant issues with decision trees is their tendency to overfit the training data. This occurs when the tree becomes too complex, capturing noise rather than the underlying pattern. Overfitting leads to poor generalization on unseen data.
- **Instability:** Small changes in the training data can lead to completely different tree structures, making decision trees sensitive to variations in the data.
- **Bias Towards Certain Features:** Decision trees can be biased towards features with more levels (categorical variables), which might not always be the most informative.

[learn Developers(2023n), Neverthoughtaboutthat(2023)]

4.4 k-Nearest Neighbors (k-NN)

k-NN is a simple yet effective classification algorithm that classifies a new data point based on the classes of the k nearest neighbors in the feature space. While it's intuitive and requires no training phase, it can be computationally intensive for large datasets, as each new point must be compared to the entire dataset.

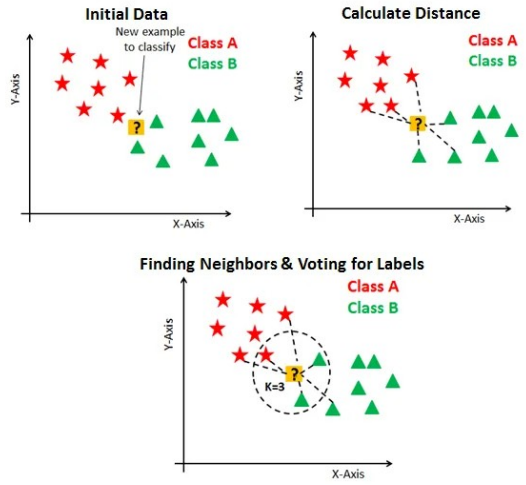

Figure 4: K-Nearest Neighbors

Advantages of k-Nearest Neighbors

- **Simplicity:** k-NN is easy to understand and implement. The concept of classifying based on the nearest neighbors is intuitive.
- **No Training Phase:** Unlike many other algorithms, k-NN does not require a training phase, which can be advantageous for certain applications where quick deployment is needed.
- **Adaptability:**The algorithm can be easily adapted to different types of data and can be used for both classification and regression tasks.

Disadvantages of k-Nearest Neighbors

- **Computationally Intensive:** As the dataset grows, the algorithm becomes slower because it needs to compute the distance to all other points in the dataset for each new instance. This can lead to significant delays in classification
- **Sensitive to Irrelevant Features:** The performance of k-NN can degrade if irrelevant features are present in the dataset, as these can distort the distance calculations.
- **Choice of k:** The choice of the number of neighbors (k) can greatly influence the performance of the algorithm. A small value of k can lead to noise influencing the result, while a large value can smooth out the boundaries between classes.

[Neverthoughtaboutthat(2023), learn Developers(2023h)]

4.5 Naive Bayes

Naive Bayes is a classification algorithm based on Bayes' theorem, which provides a way to calculate the probability of a class based on the features of a given instance. It assumes that all features are independent from each other, simplifying the computation of probabilities. This approach is particularly effective for large datasets, especially in text classification tasks such as spam filtering and sentiment analysis, due to its speed and efficiency in processing.

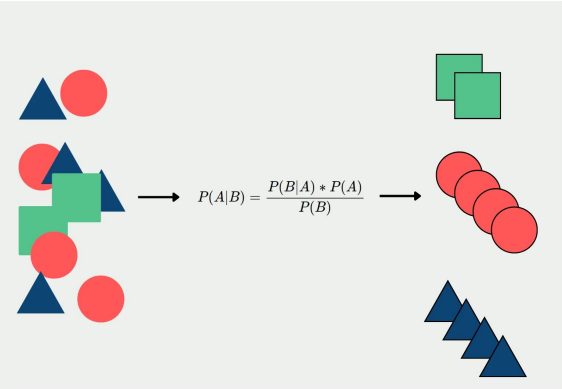

Figure 5: Naive Bayes

Advantages of Naive Bayes

- **Simplicity and Speed:** is straightforward to implement and computationally efficient. It works well with large datasets and requires minimal training time. (
- **Effective with High Dimensional Data:** It performs well in high-dimensional spaces, such as text classification, where the number of features can be very large.
- **Good Performance with Small Data:** ven with small datasets, Naive Bayes can provide good performance, making it a suitable choice for scenarios where data availability is limited.

Disadvantages of Naive Bayes

- **Assumption of Independence:** The major drawback is the assumption of independence between features, which is rarely true in real-world data. This can lead to suboptimal performance when features are correlated.
- **Limited Expressiveness:** models may be less expressive than other classifiers, which can limit their performance on more complex tasks.
- **Zero Probability Problem:** If a category has a feature that wasn't observed in the training set, it can lead to a zero probability in the model's predictions. This issue can be mitigated using techniques like Laplace smoothing.

[Neverthoughtaboutthat(2023), learn Developers(2023g), Wikipedia(2023)]

4.6 Gradient Boosting Machines (GBM)

Gradient Boosting Machines (GBM) are an ensemble machine learning technique that builds a predictive model as a sequence of weak models, typically decision trees. In GBM, each subsequent model attempts to correct the errors made by previous models. This process relies on minimizing the loss function using gradients, allowing the model to learn from the mistakes of prior predictions. Gradient boosting achieves high accuracy and is effective in many applications, such as classification and regression, but requires proper tuning to avoid overfitting.

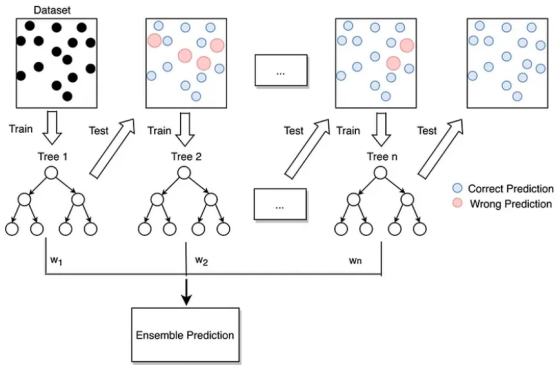

Figure 6: Gradient Boosting Machines

Advantages of Gradient Boosting Machines

- **High Accuracy:** GBM typically achieves better accuracy than individual models, especially on complex datasets.
- **Effectiveness in Diverse Applications:** GBM performs well in both classification and regression tasks, making it versatile and widely used in practice.
- **Control over Overfitting:** arameters such as the number of trees, tree depth, and learning rate allow for precise model tuning, enabling control over overfitting.

Disadvantages of Gradient Boosting Machines

- **High Computational Requirements:** Training GBM can be time-consuming and require significant computational power, especially with large datasets.
- **Hyperparameter Calibration Needed:** To achieve optimal performance, GBM requires careful calibration of hyperparameters, which can be challenging and time-consuming.
- **Prone to Overfitting:** Due to its iterative nature of learning from errors, GBM is susceptible to overfitting if not properly tuned.

[Neverthoughtaboutthat(2023), Science(2020b), learn Developers(2023c)]

5 Regression Models

Regression models are a type of supervised learning used to predict continuous outcomes based on input features. They are widely used in various fields, such as finance, biology, and social sciences, to model relationships between variables. Here is a summary of some commonly used regression models, their strengths, and limitations.

5.1 Linear Regression

Linear regression is one of the simplest and most widely used statistical techniques in machine learning and data analysis. It is a supervised learning algorithm used to model the relationship between a dependent variable (target) and one or more independent variables (features) by fitting a linear equation to the observed data.
The linear regression model can be expressed as:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + \epsilon$$

where:

- y is the predicted value (dependent variable).
- β_0 is the intercept term.
- $\beta_1, \beta_2, \ldots, \beta_n$ are the coefficients for each feature.
- $x_1, x_2, \ldots, x_n$ are the independent variables (features).
- ϵ is the random error term.

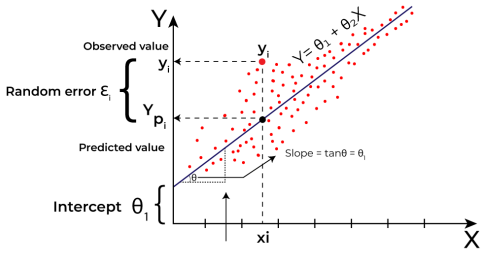

Figure 7: Linear Regression

Advantages of Linear Regression

- **Simplicity and Interpretability:** Linear regression is easy to understand and interpret. The relationship between variables can be easily visualized, and the coefficients indicate the strength and direction of the relationships.
- **Efficiency:** It requires less computational resources compared to more complex algorithms, making it suitable for small datasets or when computational efficiency is crucial.
- **Good Performance with Linearly Separable Data:** Linear regression performs well when the relationship between the dependent and independent variables is approximately linear.

Disadvantages of Linear Regression

- **Assumption of Linearity:** Linear regression assumes a linear relationship between the dependent and independent variables, which may not always be valid for complex datasets.
- **Sensitivity to Outliers:** Linear regression is sensitive to outliers, which can disproportionately influence the model's parameters and lead to inaccurate predictions.
- **Multicollinearity:** If independent variables are highly correlated, it can lead to instability in coefficient estimates and make the model unreliable.

[learn Developers(2023e)]

5.2 Ridge Regression

Ridge regression, also known as L2 regression, is an extension of linear regression that introduces regularization to the model. Regularization adds a penalty to the cost function, altering how the model fits the data. In the case of ridge regression, a penalty term proportional to the square of the coefficients is added, helping to reduce their magnitude and thus combat overfitting.
The ridge regression model can be expressed with the equation:

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n$$

where the penalty term is represented as:

$$\text{Penalty} = \lambda \sum_{j=1}^{n} \beta_j^2$$

Where:

- $\hat{y}$ is the predicted value (dependent variable).
- β_0 is the intercept.
- $\beta_1, \beta_2, \ldots, \beta_n$ are the coefficients for each feature.
- λ is the regularization parameter controlling the penalty's strength.

Advantages of Ridge Regression

- **Reduction of Overfitting:** Regularization helps reduce the risk of overfitting the model, especially in complex datasets with many features.
- **Stability of Coefficients:** By shrinking the coefficient values, ridge regression increases the stability of the model, which is particularly important when multicollinearity (high correlation between independent variables) is present.
- **Good Performance in High Dimensions:** Ridge regression performs well with datasets that have many features, even when some of them are less relevant.

Disadvantages of Ridge Regression

- **No Feature Selection:** Ridge regression does not perform feature selection, meaning it does not eliminate irrelevant variables from the model, unlike Lasso regression.
- **Parameter Tuning Required:** Selecting the appropriate regularization parameter λ is crucial for achieving optimal results, which can be difficult and time-consuming.
- **Less Effective with Strong Overfitting:** In cases where data is heavily overfitted, ridge regression may not be sufficient to achieve satisfactory results.

[learn Developers(2023k)]

5.3 Lasso Regression

Lasso regression, which stands for Least Absolute Shrinkage and Selection Operator, is a regression technique that performs both linear regression and feature selection through the application of L1 regularization. In Lasso regression, a penalty term is added to the cost function, equal to the sum of the absolute values of the coefficients. This regularization allows for the reduction (shrinkage) of coefficient values and the setting of some to zero, effectively eliminating less significant variables.
The Lasso regression model can be expressed with the equation:

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n$$

with an additional penalty term:

$$\text{Penalty} = \lambda \sum_{j=1}^{n} |\beta_j|$$

Where:

- $\hat{y}$ is the predicted value (dependent variable).
- β_0 is the intercept.
- $\beta_1, \beta_2, \ldots, \beta_n$ are the coefficients for each feature.
- λ is the regularization parameter that controls the strength of the penalty.

Advantages of Lasso Regression

- **Feature Selection:** Lasso regression effectively performs feature selection by eliminating less significant variables, which can lead to simpler models and better interpretability.
- **Reduction of Overfitting:** The introduction of regularization helps reduce the risk of overfitting, especially with complex datasets.
- **Ease of Implementation:** Lasso regression is relatively easy to implement and can be quickly applied to various regression problems.

Disadvantages of Lasso Regression

- **Unpredictability in Selection:** In cases of strong multicollinearity (high correlation among independent variables), Lasso regression may randomly select one among many correlated features, leading to instability in the results.
- **Parameter Calibration Required:** Selecting the appropriate regularization parameter λ is crucial for achieving optimal results, which can be challenging and time-consuming
- **Limitations with Nonlinear Models:** Lasso regression may not perform well with nonlinear relationships among variables, which can lead to an underestimation of the model's complexity.

[learn Developers(2023d), Science(2019a)]

5.4 Elastic Net Regression

Elastic Net Regression is a regression technique that combines the strengths of both Lasso and Ridge regression by applying both L1 and L2 regularization. It is particularly useful when there are many features in the data, and some of them are correlated. Elastic Net adds penalty terms to the cost function that are equal to the sum of the absolute values of the coefficients (L1) and the sum of the squares of the coefficients (L2). This combination allows for both feature selection and regularization, making Elastic Net a very flexible tool in data analysis.
The Elastic Net regression model can be expressed with the equation:

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n$$

with an additional penalty term:

$$\text{Penalty} = \lambda_1 \sum_{j=1}^{n} |\beta_j| + \lambda_2 \sum_{j=1}^{n} \beta_j^2$$

Where:

- $\hat{y}$ is the predicted value (dependent variable).
- β_0 is the intercept.
- $\beta_1, \beta_2, \ldots, \beta_n$ are the coefficients for each feature.
- λ_1 is the regularization parameter for L1.
- λ_2 is the regularization parameter for L2.

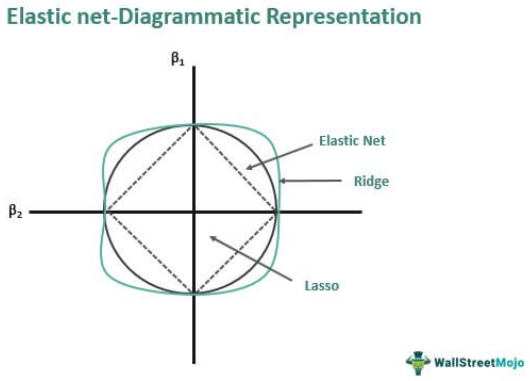

Figure 8: Elastic Net Regression

Advantages of Elastic Net Regression

- **Feature Selection:** Elastic Net effectively performs feature selection by eliminating less significant variables, which can lead to simpler models.
- **Stability in Computation:** In cases of strong multicollinearity (high correlation among variables), Elastic Net provides greater stability than Lasso.
- **Flexibility:** By combining the benefits of both Lasso and Ridge regression, Elastic Net is a flexible modeling tool that can be used in various contexts.

Disadvantages of Elastic Net Regression

- **Model Complexity:** The need to tune two regularization parameters (λ_1 and λ_2) can be challenging and time-consuming.
- **Possibility of Overfitting:** While Elastic Net helps control overfitting, there is still a risk that the model could be overly complex if parameters are poorly chosen.
- **Less Interpretability:** Elastic Net regression models can be harder to interpret compared to simpler models, which may be problematic in applications where transparency is key.

[learn Developers(2023b), Science(2020a), Vidhya(2017)]

5.5 Polynomial Regression

Polynomial Regression is an extension of linear regression that allows for the modeling of nonlinear relationships between the dependent and independent variables. Instead of fitting a straight line to the data, polynomial regression fits a polynomial equation, which can capture more complex patterns in the data.
The polynomial equation can be represented as follows:

$$\hat{y} = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + ... + \beta_n x^n$$

Where:

- $\hat{y}$ is the predicted value (dependent variable).
- β_0 is the intercept.
- $\beta_1, \beta_2, \ldots, \beta_n$ are the coefficients for each feature, where n indicates the degree of the polynomial.

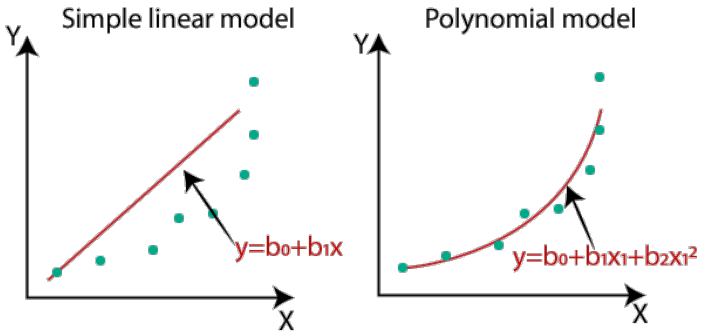

Figure 9: Polynomial Regression

Advantages of Polynomial Regression

- **Captures Nonlinear Relationships:** Polynomial regression can effectively model nonlinear relationships between the variables, making it a versatile tool for data analysis.
- **Flexible Model:** The degree of the polynomial can be adjusted to fit the complexity of the data, allowing for a more tailored approach compared to linear regression.
- **Ease of Implementation:** Polynomial regression is straightforward to implement and can be done easily with libraries like Scikit-learn in Python.

Disadvantages of Polynomial Regression

- **Risk of Overfitting:** Higher-degree polynomials can lead to overfitting, where the model captures noise in the data rather than the underlying relationship.
- **Sensitivity to Outliers:** Polynomial regression can be sensitive to outliers, which can significantly affect the shape of the fitted polynomial curve.
- **Complexity in Interpretation:** As the degree of the polynomial increases, the resulting model becomes more complex, making it harder to interpret and understand the relationship between the variables.

[Science(2019b), Vidhya(2021a), learn Developers(2023i)]

5.6 Support Vector Regression (SVR)

Support Vector Regression (SVR) is a regression technique based on the concept of Support Vector Machines (SVM). SVR aims to find a hyperplane that best fits the data while minimizing the error. Unlike traditional regression, SVR focuses on predicting values within a specified margin, making the model more robust against outliers.
In SVR, the goal is to find a function that maximizes the margin of error. It operates by determining a boundary where the prediction error does not exceed a specified threshold, known as epsilon (ϵ). In practice, SVR aims to minimize the cost function:

$$\text{Cost} = \frac{1}{2}||w||^2 + C\sum_{i=1}^{n}\xi_i$$

where:

- $||w||^2$ is the norm of the weight vector, controlling the model's complexity.
- C is the regularization parameter that penalizes errors in predictions.
- ξ_i are the slack variables representing the errors.

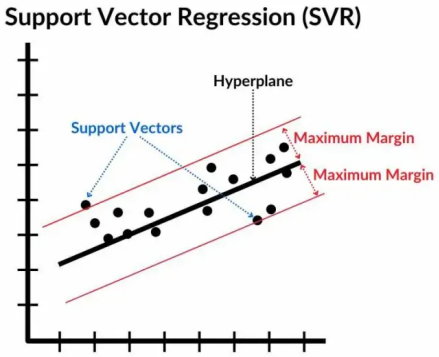

Figure 10: Support Vector Regression

Advantages of Support Vector Regression

- **Resistance to Outliers:** Unlike traditional regression techniques, SVR exhibits a higher degree of resistance to outliers present in the dataset, allowing for more reliable predictions.
- **Performance in High-Dimensional Spaces:** SVR excels when dealing with datasets characterized by a large number of features, making it particularly effective for the analysis of intricate data patterns.
- **Ability to Model Nonlinear Relationships:** By employing kernel functions, SVR has the capacity to capture and model nonlinear interactions between variables, enhancing its versatility.

Disadvantages of Support Vector Regression

- **Kernel Function Selection:** Determining the most appropriate kernel function and fine-tuning the associated model parameters can be a complex task, often requiring a trial-and-error approach.
- **High Computational Requirements:** SVR can demand significant computational resources, particularly during the training phase when working with large datasets, which may impact performance.
- **Challenges in Interpretability:** Compared to more straightforward regression models, SVR can be less interpretable, making it difficult to understand the underlying decision process of the model.

[Vidhya(2020), Science(2019c), learn Developers(2023m)]

5.7 Random Forest Regression

Random Forest Regression is an ensemble learning method that builds multiple decision trees and outputs the average prediction from these trees. This approach improves prediction stability and accuracy, especially for complex datasets with high variance.

How It Works?

- **Building Trees:** The algorithm creates numerous decision trees using random subsets of training data and features, introducing diversity among the trees.
- **Averaging Predictions:** Each tree makes a prediction for input, and the final Random Forest prediction is the average of all individual tree predictions, reducing the risk of overfitting.
- **Feature Importance:** Random Forest can assess the importance of different features in predicting the target variable, aiding in feature selection and model interpretation.

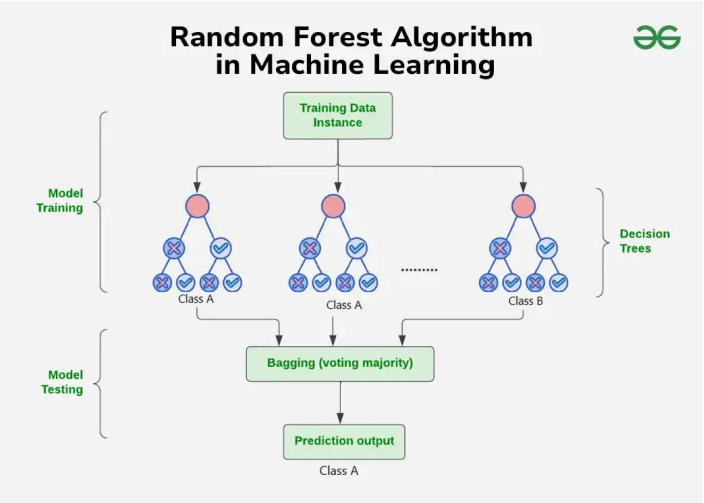

Figure 11: Radnom Forest

Advantages of Random Forest Regression

- **Reduced Overfitting:** Averages results from multiple trees to enhance generalizability to unseen data.
- **High Accuracy:** Typically achieves greater accuracy than individual decision trees, suitable for various regression tasks.
- **Robustness:** Less sensitive to noise and outliers, increasing reliability in real-world applications.

Disadvantages of Random Forest Regression

- **Complexity:** More complex and harder to interpret than a single decision tree, making individual contributions challenging to understand.
- **Longer Training Times:** Training multiple trees can be significantly slower, especially with large datasets.
- **Memory Usage:** High memory consumption due to the storage of multiple trees can be an issue in resource-limited environments.

[Neverthoughtaboutthat(2023), learn Developers(2023j), Vidhya(2021b)]

6 Ensemble Methods

Ensemble methods are powerful techniques in machine learning that combine multiple models to improve overall performance and robustness. By aggregating the predictions of various models, ensemble methods can reduce errors, enhance accuracy, and mitigate overfitting. Common ensemble techniques include:

- **Random Forest:** An ensemble of decision trees that improves accuracy and robustness by averaging predictions and reducing overfitting.
- **AdaBoost:** A boosting algorithm that combines weak learners to create a strong classifier, focusing on the misclassified instances from previous iterations.
- **Gradient Boosting:** An iterative approach that builds models sequentially, each correcting the errors of the previous one. Variants like XGBoost, LightGBM, and CatBoost offer enhanced performance and flexibility.

[learn Developers(2023a)]

7 Conclusion

Machine learning encompasses a diverse range of algorithms and techniques for analyzing data and making predictions. Understanding different models, such as regression and ensemble methods, allows practitioners to select the most appropriate approach based on the specific problem at hand. As the field continues to evolve, ongoing research and development will further enhance the capabilities and applications of machine learning in various domains.

References

[Academy(2024)] EITCA Academy. What is machine learning?, 2024. URL https://pl.eitca.org/sztuczna-inteligencja/eitc-ai-gcml-Google-Cloud-Machine-Learning/wprowadzenie/czym-jest-uczenie-maszynowe/co-to-jest-ml/.

[Codecademy(2023)] Codecademy. Exploratory data analysis: Data visualization techniques, 2023. URL https://www.codecademy.com/article/eda-data-visualization.

[Combeenut.pl(2023)] Redakcja Combeenut.pl. Machine learning: definitions, methods and applications in practice, 2023. URL https://combeenut.pl/sztuczna-inteligencja-i-machine-learning/machine-learning/.

[GeeksforGeeks(2023a)] GeeksforGeeks. Splitting data for machine learning models, 2023a. URL https://www.geeksforgeeks.org/splitting-data-for-machine-learning-models/.

[GeeksforGeeks(2023b)] GeeksforGeeks. Features and labels in supervised learning: A practical approach, 2023b. URL https://www.geeksforgeeks.org/features-and-labels-in-supervised-learning-a-practical-approach/.

[learn Developers(2023a)] Scikit learn Developers. Adaboostregressor in scikit-learn, 2023a. URL https://scikit-learn.org/1.5/modules/generated/sklearn.ensemble.AdaBoostRegressor.html.

[learn Developers(2023b)] Scikit learn Developers. Elastic net in scikit-learn, 2023b. URL https://scikit-learn.org/stable/modules/linear_model.html.

[learn Developers(2023c)] Scikit learn Developers. Gradient boosting in scikit-learn, 2023c. URL https://scikit-learn.org/stable/modules/ensemble.html.

[learn Developers(2023d)] Scikit learn Developers. Lasso regression in scikit-learn, 2023d. URL https://scikit-learn.org/stable/modules/linear_model.html.

[learn Developers(2023e)] Scikit learn Developers. Linear regression in scikit-learn, 2023e. URL https://scikit-learn.org/stable/modules/linear_model.html.

[learn Developers(2023f)] Scikit learn Developers. Logistic regression in scikit-learn, 2023f. URL https://scikit-learn.org/stable/modules/linear_model.html.

[learn Developers(2023g)] Scikit learn Developers. Naive bayes in scikit-learn, 2023g. URL https://scikit-learn.org/stable/modules/naive_bayes.html.

[learn Developers(2023h)] Scikit learn Developers. Nearest neighbors in scikit-learn, 2023h. URL https://scikit-learn.org/stable/modules/neighbors.html.

[learn Developers(2023i)] Scikit learn Developers. Polynomial regression in scikit-learn, 2023i. URL https://scikit-learn.org/stable/modules/linear_model.html.

[learn Developers(2023j)] Scikit learn Developers. Random forests in scikit-learn, 2023j. URL https://scikit-learn.org/stable/modules/ensemble.html.

[learn Developers(2023k)] Scikit learn Developers. Ridge regression in scikit-learn, 2023k. URL https://scikit-learn.org/stable/modules/linear_model.html.

[learn Developers(2023l)] Scikit learn Developers. Support vector machines (svm) in scikit-learn, 2023l. URL https://scikit-learn.org/stable/modules/svm.html.

[learn Developers(2023m)] Scikit learn Developers. Support vector regression (svr) in scikit-learn, 2023m. URL https://scikit-learn.org/stable/modules/svm.html.

[learn Developers(2023n)] Scikit learn Developers. Decision trees in scikit-learn, 2023n. URL https://scikit-learn.org/stable/modules/tree.html.

[MachineLearningPlus(2023)] MachineLearningPlus. Exploratory data analysis (eda) in python: A practical guide, 2023. URL https://www.machinelearningplus.com/machine-learning/exploratory-data-analysis-eda/.

[Microsoft(2023)] Microsoft. Exploratory data analysis in ai: Capabilities and experimentation, 2023. URL https://learn.microsoft.com/en-us/ai/playbook/capabilities/experimentation/exploratory-data-analysis.

[Neverthoughtaboutthat(2023)] Neverthoughtaboutthat. Classification of machine learning algorithms for beginners, 2023. URL https://neverthoughtaboutthat.com/pl/klasyfikacja-algorytmÃşw-uczenia-maszynowego-dla-poczÄĚtkujÄĚcych/#SVM_Support_Vector_Machine.

[Science(2019a)] Towards Data Science. Lasso regression tutorial, 2019a. URL https://towardsdatascience.com/lasso-regression-tutorial-fd68de0aa2a2.

[Science(2019b)] Towards Data Science. Polynomial regression in python, 2019b. URL https://towardsdatascience.com/polynomial-regression-in-python-dd655a7d9f2b.

[Science(2019c)] Towards Data Science. Machine learning basics: Support vector regression, 2019c. URL https://towardsdatascience.com/machine-learning-basics-support-vector-regression-660306ac5226.

[Science(2020a)] Towards Data Science. How to use elastic net regression, 2020a. URL https://towardsdatascience.com/how-to-use-elastic-net-regression-85a6a393222b.

[Science(2020b)] Towards Data Science. Understanding gradient boosting machines, 2020b. URL https://towardsdatascience.com/understanding-gradient-boosting-machines-9be756fe76ab.

[Talent500(2023)] Talent500. The role of exploratory data analysis in machine learning, 2023. URL https://talent500.co/blog/role-of-exploratory-data-analysis-in-machine-learning/.

[Vidhya(2017)] Analytics Vidhya. A comprehensive guide for linear, ridge and lasso regression, 2017. URL https://www.analyticsvidhya.com/blog/2017/06/a-comprehensive-guide-for-linear-ridge-and-lasso-regression/.

[Vidhya(2020)] Analytics Vidhya. Support vector regression tutorial for machine learning, 2020. URL https://www.analyticsvidhya.com/blog/2020/03/support-vector-regression-tutorial-for-machine-learning/.

[Vidhya(2021a)] Analytics Vidhya. Understanding polynomial regression model, 2021a. URL https://www.analyticsvidhya.com/blog/2021/10/understanding-polynomial-regression-model/.

[Vidhya(2021b)] Analytics Vidhya. Understanding random forest, 2021b. URL https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/.

[Wikipedia(2023)] Wikipedia. Naive bayes classifier, 2023. URL https://en.wikipedia.org/wiki/Naive_Bayes_classifier.