



New Thread

Answer Profile

yes?

find latest technical paper on blockchain tech which discuss inn

find me researchch

Clear Chats

Try Phind Pro

Pair Programmer Use Fast Model

[Upgrade for more daily best model uses](#)

What is impurity measure in decision trees?

ANSWER | GPT-4 MODEL

Impurity measure in decision trees is a metric that quantifies the homogeneity of labels at a particular node. It helps determine the best attribute for splitting the data at each node in a decision tree. The lower the impurity, the more homogeneous (or "pure") the node is. Impurity is commonly measured using the Gini Impurity Index and Entropy[0][1].

1. Gini Impurity: The Gini Index or Gini Impurity measures the degree or probability of a particular variable being wrongly classified when it is randomly picked. If all the elements in a node fall into a single category, the Gini impurity is 0 (indicating a pure node). If the elements are randomly distributed across various categories, the Gini impurity is 0.5 (for a binary classification)[0][3][5][6].

The Gini Impurity for a node is calculated as:

$$\text{Gini Index} = 1 - \sum (p_i)^2$$



where p_i is the probability of an object being classified to a particular class [3][5][6].

2. Entropy: Entropy is another measure of impurity, disorder, or uncertainty. The entropy of a node increases with the increase of impurity or disorder. If the sample is completely homogeneous, the entropy is zero, and if the sample is equally divided, it has entropy of one[0][6].

Entropy $H(S)$ for a given set S is calculated as:

$$H(S) = - \sum p_i * \log_2(p_i)$$



where p_i is the proportion (or probability) of the elements that are in class i in the set S [3][6].

The goal in decision tree learning is to minimize the impurity in the nodes. The decision tree algorithm does this by choosing the attribute that results in the most homogeneous child nodes for each split, i.e., the attribute that provides the most information gain. Information gain is calculated based on the decrease in entropy after a dataset is split on an attribute. Constructing a decision tree is all about finding attribute that returns the highest information gain[3][6].

While both Gini Impurity and Entropy are commonly used in decision tree algorithms, the choice between the two often depends on the specific problem and requirements. In general, they tend to generate similar trees. Gini impurity tends to be faster to compute, so it is a good default choice. However, when the attributes are continuous, entropy can sometimes produce more balanced trees[6].

SOURCES

[www.baeldung.com > cs > impurit...](#)

[Node Impurity in Decision Trees | Baeldung on Computer Science](#)

Informally, impurity is a measure of homogeneity of the labels at the node at hand: There are different ways to define impurity. In classification tasks,..

[stats.stackexchange.com > questi...](#)

[What is node impurity/purity in decision trees? – Cross Validated](#)

What is node impurity/purity in decision trees? Classification Trees Intuitively, you can think of a set of examples as the set of atoms in a..

[towardsdatascience.com > impuri...](#)

[Impurity & Judging Splits – How a Decision Tree Works](#)

In essence a Decision Tree is a flow diagram where it asks a series of questions about a data point (which has a set of..

[en.wikipedia.org > wiki > Decision...](#)

[Decision tree learning – Wikipedia](#)

Decision tree learning is a supervised learning approach used in statistics, data mining and machine learning. In this formalism, a classification or regression...

[www.cmi.ac.in > ~madhavan > cou...](#)

[Lecture 7: Impurity Measures for Decision Trees](#)

Madhavan Mukund
<https://www.cmi.ac.in/~madhavan> Data Mining and Machine Learning August{December 2020 Misclassification rate Goal: partition with uniform categor..

[0] baeldung.com

[1] stats.stackexchange.com

[3] en.wikipedia.org

[5] towardsdatascience.com

[6] geeksforgeeks.org

How to measure impurity (decision tree)?

ANSWER | GPT-4 MODEL

Impurity measure in decision trees is a metric that quantifies the homogeneity of labels at a particular node. It helps determine the best attribute for splitting the data at each node in a decision tree. The lower the impurity, the more homogeneous (or "pure") the node is. Impurity is commonly measured using the Gini Impurity Index and Entropy[0][1].

1. Gini Impurity: The Gini Index or Gini Impurity measures the degree or probability of a particular variable being wrongly classified when it is randomly picked. If all the elements in a node fall into a single category, the Gini impurity is 0 (indicating a pure node). If the elements are randomly distributed across various categories, the Gini impurity is 0.5 (for a binary classification)[0][3][5][6].

The Gini Impurity for a node is calculated as:

$$\text{Gini Index} = 1 - \sum (p_i)^2$$



where p_i is the probability of an object being classified to a particular class [3][5][6].

2. Entropy: Entropy is another measure of impurity, disorder, or uncertainty. The entropy of a node increases with the increase of impurity or disorder. If the sample is completely homogeneous, the entropy is zero, and if the sample is equally divided, it has entropy of one[0][6].

Entropy $H(S)$ for a given set S is calculated as:

$$H(S) = - \sum p_i * \log_2(p_i)$$



where p_i is the proportion (or probability) of the elements that are in class i in the set S [3][6].

The goal in decision tree learning is to minimize the impurity in the nodes. The decision tree algorithm does this by choosing the attribute that results in the most homogeneous child nodes for each split, i.e., the attribute that provides the most information gain. Information gain is calculated based on the decrease in entropy after a dataset is split on an attribute. Constructing a decision tree is all about finding attribute that returns the highest information gain[3][6].

While both Gini Impurity and Entropy are commonly used in decision tree algorithms, the choice between the two often depends on the specific problem and requirements. In general, they tend to generate similar trees. Gini impurity tends to be faster to compute, so it is a good default choice. However, when the attributes are continuous, entropy can sometimes produce more balanced trees[6].

decision tree impurity measure calculation

SOURCES ↗

www.baeldung.com > cs > impurit...

Node Impurity in Decision Trees | Baeldung on Computer Science

1. Introduction In this tutorial, we'll talk about node impurity in decision trees. A decision tree is a greedy algorithm we use for supervised machine..

towardsdatascience.com > impuri...

Impurity & Judging Splits – How a Decision Tree Works

In essence a Decision Tree is a flow diagram where it asks a series of question about a data point (which has a set of..

www.cmi.ac.in > ~madhavan > cou...

Lecture 7: Impurity Measures for Decision Trees – CMI

= Entropy Information theoretic measure of randomness Minimum number of bits to transmit message | [Shannon] data items nO with c = 0, p0..

www.geeksforgeeks.org > gini-imp...

ML | Gini Impurity and Entropy in Decision Tree – GeeksforGeeks

The Gini Index, also known as Impurity, calculates the likelihood that somehow a randomly picked instance would be erroneously cataloged. Machine Learning is a ..

towardsdatascience.com > gini-i...

Gini Impurity Measure – a simple explanation using python

The Gini impurity measure is one of the methods used in decision tree algorithms to decide the optimal split from a root node, and subsequent..

CITATIONS

[0] baeldung.com

[1] towardsdatascience.com

[3] geeksforgeeks.org

[5] stats.stackexchange.com

[6] people.revoledu.com

Why we should measure impurity in decision trees?

ANSWER | GPT-4 MODEL

Impurity measures in decision trees are crucial for determining the best attribute for splitting the data at each node in a decision tree. The goal is to minimize the impurity in the nodes, meaning that we want each node to be as homogeneous as possible with respect to the target variable. By measuring impurity, we can construct a decision tree that is more effective at classifying or predicting new instances [0][1].

Impurity measures help us decide how to split the data at each node in a decision tree. When constructing a decision tree, the algorithm will consider each attribute and compute the impurity of the child nodes that would result from splitting on that attribute. The attribute that results in the smallest impurity is chosen for the split [0][1].

Impurity measures are important because they allow the decision tree algorithm to make informed decisions about how to split the data at each node. By minimizing impurity, we are maximizing the homogeneity of the target variable within each node. This leads to more accurate predictions when the decision tree is used on new data [0][1].

Another reason why we measure impurity is to prevent overfitting. Overfitting happens when the model learns the training data too well, to the point where it performs poorly on unseen data. By using impurity measures to decide when to stop splitting the data (i.e., when the impurity is low enough), we can create a model that generalizes well to unseen data [2].

In conclusion, measuring impurity in decision trees is fundamental to the construction of the tree, it helps to decide the best attribute for splitting at each node, and it prevents overfitting. The two common measures of impurity are Gini Impurity and Entropy. Both are effective, and the choice between the two often depends on the specific problem and requirements [0][1][6].

CITATIONS

[0] baeldung.com

[1] towardsdatascience.com

[2] stats.stackexchange.com

[6] geeksforgeeks.org

What is information gain? How it is useful in selecting decision node?

reasons for measuring impurity in decision trees

SOURCES ↗

🔗 www.baeldung.com > cs > impurit...

[Node Impurity in Decision Trees | Baeldung on Computer Science](#)

Introduction In this tutorial, we'll talk about node impurity in decision trees. A decision tree is a greedy algorithm we use for supervised machine learning..

🔗 towardsdatascience.com > impuri...

[Impurity & Judging Splits — How a Decision Tree Works](#)

The reduction in impurity is the starting group Gini impurity minus the weighted sum of impurities from the resulting split groups. This is 0.3648–0.2747 =..

🔗 stats.stackexchange.com > questi...

[What is node impurity/purity in decision trees? – Cross Validated](#)

1 Answer Sorted by: 9 What is node impurity/purity in decision trees? Classification Trees Intuitively, you can think of a set of examples as the..

🔗 www.cmi.ac.in > ~madhavan > cou...

[Lecture 7: Impurity Measures for Decision Trees – CMI](#)

Measure of unequal distribution of wealth Economics [(Corrado Gini)] As before, n data items n O with c = 0, p O = n O=n..

🔗 datascience.stackexchange.com ... [classification – Gini impurity in decision tree \(reasons to use it ...\)](#)

3 In a decision tree, Gini Impurity [1] is a metric to estimate how much a node contains different classes. It measures the probability of..

information gain decision tree node selection

ANSWER | GPT-4 MODEL

Information gain is a key concept in machine learning, specifically in the construction of decision trees. It measures how much information a feature provides about the class, and it is used to determine the order of attributes in the nodes of a decision tree[0][4].

In the context of decision trees, information gain is the reduction in entropy or surprise by splitting a dataset according to a given value of a random variable. A larger information gain suggests a lower entropy group or groups of samples, and hence less surprise[6].

The information gain IG can be calculated as:

$$IG = \text{Entropy}(\text{Parent}) - [\text{Weighted Average}] \text{Entropy}(\text{Children})$$



where $\text{Entropy}(\text{Parent})$ represents the entropy of the parent node and $[\text{Weighted Average}] \text{Entropy}(\text{Children})$ represents the average entropy of the child nodes, weighted by the number of instances in each child node[4][6].

Entropy, in this context, is a measure of impurity or uncertainty. For a binary classification problem, the entropy of a data sample can be calculated as follows:

$$\text{Entropy} = - \sum p_i * \log_2(p_i)$$



where p_i is the proportion (or probability) of the elements that are in class i in the set S [3][6].

Information gain is the basic criterion to decide whether a feature should be used to split a node or not. The feature with the optimal split i.e., the highest value of information gain at a node of a decision tree is used as the feature for splitting the node. This approach usually leads to more homogeneous groups, making the decision tree more effective in classifying or predicting new instances[0][3].

However, while information gain is a good measure for deciding the relevance of an attribute, it is not perfect. It tends to favor attributes with a large number of distinct values. To counter this problem, one can choose the attribute with the highest information gain ratio from among the attributes whose information gain is average or higher. This biases the decision tree against considering attributes with a large number of distinct values[0].

In Python's scikit-learn library, the decision tree algorithm can be configured to use entropy as a measure for the quality of a split by setting the `criterion` argument to `"entropy"`:

```
from sklearn.tree import DecisionTreeClassifier  
model = DecisionTreeClassifier(criterion='entropy')
```



This configuration causes the model to use information gain when deciding how to split the data at each node[6].

CITATIONS

[0] en.wikipedia.org

[3] analyticsvidhya.com

[4] section.io

[6] machinelearningmastery.com

SOURCES ↗

w en.wikipedia.org > wiki > Informati...

Information gain (decision tree) – Wikipedia

We can use information gain to determine how good the splitting of nodes is in a decision tree...

⌚ towardsdatascience.com > decisi...

Decision Trees Explained – Entropy, Information Gain, Gini Index, CCP ...

Decision Trees Explained – Entropy, Information Gain, Gini Index, CCP Pruning | by Shailey Dash | Towards Data Science 500 Apologies, but something went wrong..

⌚ towardsdatascience.com > entro...

Entropy and Information Gain in Decision Trees

Entropy and Information Gain in Decision Trees A simple look at some key Information Theory concepts and how to use them when building a Decision..

↗ www.analyticsvidhya.com > blog >...

Information Gain | Best Split in Decision Trees using Information Gain

The information gain can be calculated as: But wait, what is Entropy here? How can we calculate that? Let's look at how we can do..

⌚ www.section.io > engineering-edu...

Entropy and Information Gain to Build Decision Trees in Machine ...

A decision tree is a supervised learning algorithm used for both classification and regression problems. Simply put, it takes the form of a tree with..

1 2

Let's assume we want to play badminton on a particular day – say Saturday – how will you decide whether to play or not. Create decision tree. (Example...)

ANSWER | GPT-4 MODEL

To create a decision tree for the given problem, we first need to understand the concept of decision tree building, which involves choosing the best feature for splitting at each node based on some criterion. The most common criteria are Information Gain, Gini Impurity, and Variance Reduction. In this case, we'll use Information Gain for simplicity.

Here are the given data points:

Day Weather Temperature Humidity Wind Play?

1	Sunny	Hot	High	Weak	No
2	Cloudy	Hot	High	Weak	Yes
3	Sunny	Mild	Normal	Strong	Yes
4	Cloudy	Mild	High	Strong	Yes
5	Rainy	Mild	High	Strong	No
6	Rainy	Cool	Normal	Strong	No
7	Rainy	Mild	High	Weak	Yes
8	Sunny	Hot	High	Strong	No
9	Cloudy	Hot	Normal	Weak	Yes
10	Rainy	Mild	High	Strong	No

To decide whether to play or not, we can create a decision tree model using Python's scikit-learn library:

```
import pandas as pd
from sklearn.tree import DecisionTreeClassifier
from sklearn import tree

# Create DataFrame
data = {'Weather': ['Sunny', 'Cloudy', 'Sunny', 'Cloudy', 'Rainy',
                    'Temperature': ['Hot', 'Hot', 'Mild', 'Mild', 'Mild'],
                    'Humidity': ['High', 'High', 'Normal', 'High', 'High'],
                    'Wind': ['Weak', 'Weak', 'Strong', 'Strong', 'Strong'],
                    'Play': ['No', 'Yes', 'Yes', 'Yes', 'No', 'No', 'Yes', 'Yes']}
df = pd.DataFrame(data)

# Convert categorical variable into dummy/indicator variables or
df_encoded = pd.get_dummies(df, columns=['Weather', 'Temperature'])

# Separate features from target
X = df_encoded.drop('Play', axis=1)
y = df_encoded['Play']

# Create Decision Tree classifier
clf = DecisionTreeClassifier()
clf = clf.fit(X, y)

# Visualize decision tree
tree.plot_tree(clf)
```

In this code, we first create a DataFrame from the given data, then convert categorical variables into binary variables, which is necessary for the decision tree algorithm. We separate the features from the target, then create a Decision Tree classifier and fit the model with the data. Finally, we visualize the decision tree.

Please note that this is a simplified version and real-world data would require additional steps such as data preprocessing, feature engineering, and model validation[4][7].

decision tree algorithm for classification

SOURCES ↗

▼ www.javatpoint.com > machine-le...

Decision Tree Algorithm in Machine Learning – Javatpoint

Decision Tree Classification Algorithm. Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but m..

W en.wikipedia.org > wiki > Decision...

Decision tree learning – Wikipedia

Decision tree learning is a supervised learning approach used in statistics, data mining and machine learning. In this formalism, a classification or regression...

► scikit-learn.org > stable > module...

1.10. Decision Trees — scikit-learn 1.3.0 documentation

Decision Trees (DTs) are a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the val..

© www.coursera.org > articles > dec...

Decision Trees in Machine Learning: Two Types (+ Examples)

Data Decision Trees in Machine Learning: Two Types (+ Examples) Decision Trees in Machine Learning: Two Types (+ Examples) Written by Coursera • Updated on..

► towardsdatascience.com > decis...

Decision Trees for Classification — Complete Example

Jan 13 Photo by Fabrice Villard on Unsplash This article explains how we can use decision trees for classification problems. After explaining important terms...

[4] towardsdatascience.com

[7] hackerearth.com

ANSWER | GPT-4 MODEL

Impurity measure in decision trees is a metric that quantifies the homogeneity of labels at a particular node. It helps determine... ✓

ANSWER | GPT-4 MODEL

Impurity measure in decision trees is a metric that quantifies the homogeneity of labels at a particular node. It helps determine the best attribute for splitting the data at each node in a decision tree. The lower the impurity, the more homogeneous (or "pure") the node is. Impurity is commonly measured using the Gini Impurity Index and Entropy [0][1].

1. **Gini Impurity:** The Gini Index or Gini Impurity measures the degree or probability of a particular variable being wrongly classified when it is randomly picked. If all the elements in a node fall into a single category, the Gini impurity is 0 (indicating a pure node). If the elements are randomly distributed across various categories, the Gini impurity is 0.5 (for a binary classification)[0][3][5][6].

The Gini Impurity for a node is calculated as:

$$\text{Gini Index} = 1 - \sum (p_i)^2$$



where p_i is the probability of an object being classified to a particular class [3][5][6].

2. Entropy: Entropy is another measure of impurity, disorder, or uncertainty. The entropy of a node increases with the increase of impurity or disorder. If the sample is completely homogeneous, the entropy is zero, and if the sample is equally divided, it has entropy of one[0][6].

Entropy $H(s)$ for a given set s is calculated as:

$$H(S) = - \sum p_i * \log_2(p_i)$$



where $p_{i,j}$ is the proportion (or probability) of the elements that are in class i in the set $S[3][6]$.

The goal in decision tree learning is to minimize the impurity in the nodes. The decision tree algorithm does this by choosing the attribute that results in the most homogeneous child nodes for each split, i.e., the attribute that provides the most information gain. Information gain is calculated based on the decrease in entropy after a dataset is split on an attribute. Constructing a decision tree is all about finding attribute that returns the highest information gain[3][6].

While both Gini Impurity and Entropy are commonly used in decision tree algorithms, the choice between the two often depends on the specific problem and requirements. In general, they tend to generate similar trees. Gini impurity tends to be faster to compute, so it is a good default choice. However, when the attributes are continuous, entropy can sometimes produce more balanced trees[6].

impurity measure decision
tree gini entropy

SOURCES

 www.geeksforgeeks.org > gini-imp....

ML | Gini Impurity and Entropy
in Decision Tree –

GeeksforGeeks

Entropy can be defined as a measure of the purity of the sub-split. Entropy always lies between 0 to 1. The entropy of any split..

Decision Trees Explained – Entropy, Information Gain, Gini Index, CCP ...

entropy is a measure of disorder or impurity in a node. Thus, a node with more variable composition, such..

Node Impurity in Decision Trees | Baeldung on Computer Science

Informally, impurity is a measure of homogeneity of the labels at the node at hand: There are different ways to define impurity. In classification tasks,..

When should I use Gini Impurity as opposed to Information Gain (Entropy)?

Gini impurity and Information Gain Entropy are pretty much the same. And people do use the values interchangeably. Below are the formulae of both: Gini:Gini(E)..

 stats.stackexchange.com > quest...
Entropy Impurity, Gini
Impurity, Information gain –
differences?

[0] geeksforgeeks.org

[1] towardsdatascience.com

[3] datascience.stackexchange.com

[5] en.wikipedia.org

[6] mathwarehouse.com

What is hyperplane? How will you find model parameters in case of multivariate system?

ANSWER | GPT-4 MODEL

A hyperplane is a subspace whose dimension is one less than that of its original space. For example, in a three-dimensional space, a hyperplane is a two-dimensional plane. In a two-dimensional space, a hyperplane is a one-dimensional line. Although it's hard to visualize a hyperplane in a space with four or more dimensions, the concept still applies[3][9][10].

Mathematically, a hyperplane can be defined using the equation:

$$\beta_0 + \beta_1 * X_1 + \beta_2 * X_2 + \dots + \beta_p * X_p = 0$$



Any point $X = (X_1, X_2, \dots, X_p)$ in the p -dimensional space that satisfies this equation defines a $(p - 1)$ -dimensional hyperplane[3].

In machine learning, hyperplanes are often used to separate data for classification. For example, in a Support Vector Machine (SVM), a hyperplane is learned that separates the data space into two regions for binary classification. If the dimensionality of the data set is greater than two, this process is performed multiple times for multi-way classification[10].

In case of a multivariate system, the model parameters are often found by solving an optimization problem. For instance, in SVMs, the hyperplane parameters are defined by the active, binding constraints, which correspond to the nonzero Lagrange multipliers, also known as the support vectors. The optimization problem is typically formulated in the dual form, which can be solved faster and more efficiently. Even if the primal problem is not convex, the dual problem will always have a unique optimal solution[5].

To find the model parameters, you typically use an optimization algorithm such as gradient descent or a more advanced method like Newton's method. These algorithms iteratively adjust the model parameters to minimize a cost function, which measures the discrepancy between the model's predictions and the actual data.

In Python, libraries like scikit-learn and TensorFlow can be used to solve these optimization problems and find the model parameters. For example, to train a SVM with scikit-learn, you would use code similar to the following:

```
from sklearn import svm

# X is your input data, y are your targets
clf = svm.SVC(kernel = 'linear')
clf.fit(X, y)

# The coefficients of the hyperplane can be accessed with
coef = clf.coef_
```



hyperplane multivariate system model parameters

SOURCES ↗

stackoverflow.com > questions > ...

r – Plotting data from an svm fit – hyperplane – Stack Overflow

Call: svm (formula = q ~ ., data = data, kernel = "linear") Parameters: SVM-Type: C-classification SVM-Kernel: linear cost: 1 gamma: 0.333333 Number of Support..

www.sciencedirect.com > science...

A note on simulating hyperplane-truncated multivariate normal ...

1. Introduction The simulation of conditional multivariate normal (MVN) distributions with linear equality constraints is widely used in many spatial statistics..

research.monash.edu > en > publi...

Bayesian Multi-Hyperplane Machine for pattern recognition

1 Citation (Scopus) Overview Abstract Current existing multi-hyperplane machine approach deals with high-dimensional and complex datasets by approximating the i..

link.springer.com > chapter > 10.10...

Support Vector Machines and Support Vector Regression

A hyperplane is a subspace whose dimension (cardinality) is one less than that of its original space.This means that the hyperplane of a p -dimensional space..

link.springer.com > article > 10.10...

Optimal arrangements of hyperplanes for SVM-based multiclass ... – Springer

In this paper, we present a novel SVM-based approach to construct multiclass classifiers by means of arrangements of hyperplanes. We propose different mixed int..

In this code, `svm.SVC` creates a SVM classifier with a linear kernel, `fit` trains the classifier, and `coef_` contains the coefficients of the hyperplane[5][10].

CITATIONS

[3] link.springer.com

[5] link.springer.com

[9] math.libretexts.org

[10] deepai.org

How logistic regression differs from linear regression?

When you will use logistic regression?

ANSWER | GPT-4 MODEL

Logistic regression and linear regression are both statistical methods used for the prediction of dependent variables from a set of predictor variables. However, they are used for different types of problems and have different assumptions and methodologies[0][2][3][5][8]:

Linear regression is used when the dependent variable is continuous. It models the relationship between the dependent variable and the independent variables by fitting a linear equation to observed data. The steps to perform linear regression are simpler and faster, which makes it a good fit for problems with a large number of features[0][2][3][5][8].

Logistic regression, on the other hand, is used when the dependent variable is categorical, often for binary classification problems. Rather than fitting a line to the data, logistic regression fits an "S"-shaped logistic function, which can range from 0 to 1. Therefore, logistic regression predicts the probability of the outcome[0][2][3][5][8].

In terms of mathematical representation, linear regression uses the ordinary least squares method to find the best-fitting line, while logistic regression uses the maximum likelihood estimation method to find the best-fitting sigmoid curve[0][2][3][5][8].

Here are some situations when you would use logistic regression:

1. **Predicting a binary outcome:** For example, predicting whether an email is spam (1) or not spam (0), whether a tumor is malignant (1) or not (0), or whether a customer will churn (1) or not (0)[0][2][3][5][8].
2. **Predicting probabilities:** If you want to have a probabilistic understanding of your model's predictions, logistic regression should be your go-to method. For example, if you are building a model to predict customer churn, it might be beneficial to know not just whether a customer will churn (1) or not (0), but also the probability of the customer churning[0][2][3][5][8].

In conclusion, while linear regression and logistic regression are similar in that they are both used for prediction and are based on the relationship between the dependent and independent variables, they are used in different types of problems and have different methodologies and assumptions. The key is to understand these differences and choose the right method based on the type of problem you are dealing with.

CITATIONS

[0] statology.org

[2] analyticsvidhya.com

[3] indeed.com

[5] geeksforgeeks.org

[8] aws.amazon.com

logistic regression vs linear regression use cases

SOURCES ↗

🔗 www.statology.org > logistic-regre...

Logistic Regression vs. Linear Regression: The Key Differences

Difference #1: Type of Response
Variable A linear regression model is used when the response variable takes on a continuous value such as: Price Height..

🔗 datascience.stackexchange.com ...

When to use Linear Regression and When to use Logistic regression – use ...

When to use Linear Regression and When to use Logistic regression – use cases Ask Question Asked 5 years, 9 months ago Modified 5 years...

↗ www.analyticsvidhya.com > blog >...

Regression Analysis | Logistic Regression Vs Linear Regression

Linear vs Logistic Regression – Use Cases Conclusion Frequently Asked Questions Use of Regression Analysis Before starting lets quickly understand the use of re..

ℹ www.indeed.com > career-advice...

Logistic Regression vs. Linear Regression: Key Differences

While logistic regression helps classify computational problems, linear models calculate the regression line of a problem. The two types of linear regression ar..

🔗 datascience.stackexchange.com ...

When to use linear or logistic regression?

2 Answers Sorted by: 5 Linear Regression is used for predicting continuous variables. Logistic Regression is used for predicting variables which has only limite..

A purely autonomous car to be designed so that it will drive itself on roads. Define this problem as machine learning problem.

ANSWER | GPT-4 MODEL

Designing a purely autonomous car that can drive itself on roads is a complex machine learning problem. It involves various sub-problems that need to be solved, each of which can be defined as an individual machine learning problem[6][10]:

1. Perception: This is the ability of the car to perceive and understand its environment. This involves detecting and recognizing objects around the car, such as other vehicles, pedestrians, traffic signs, and lanes. This can be formulated as a supervised learning problem, where the goal is to classify different objects based on sensor data[6][10].
2. Prediction: Once the car understands its environment, it needs to predict the future actions of the objects around it. For instance, the car needs to predict whether a pedestrian will cross the street or if a car will change lanes. This can be viewed as a time-series prediction problem, where the past observations are used to predict future actions[6][10].
3. Planning: Based on the perception and prediction, the car needs to plan its actions. This involves deciding when to change lanes, when to overtake, when to stop, etc. This can be expressed as a reinforcement learning problem, where the car learns to take actions based on the rewards it receives[6][10].
4. Control: Finally, the car needs to execute the planned actions. This involves controlling the steering, acceleration, and brakes. This can be formulated as a supervised learning problem, where the inputs are the desired state and the outputs are the control commands for the actuators[6][10].

Each of these sub-problems involves processing large amounts of data from various sensors like cameras, lidar, radar, and ultrasonic sensors. Deep learning techniques are often used due to their ability to handle high-dimensional data and learn complex patterns. Moreover, these problems need to be solved in real-time, which poses additional challenges in terms of computational resources and algorithm efficiency[6][10].

It's important to note that the success of the machine learning model in driving the car autonomously will depend not only on the individual performance of each sub-problem but also on how well these sub-problems are integrated together. This is a complex task that requires a deep understanding of machine learning algorithms as well as automotive engineering[6][10].

CITATIONS

[6] en.wikipedia.org

[10] synopsys.com

How will you choose learning experience for a machine learning system?

ANSWER | GPT-4 MODEL

Choosing the learning experience for a machine learning system involves several steps and considerations, as outlined in the sources. Here are the steps to follow:

1 Defining the Problem: Clearly defining the task that the machine

learning approach for autonomous car driving

SOURCES ↗

E www.sciencedirect.com > science...

Autonomous Driving

Architectures: Insights of Machine Learning and Deep ...

Abstract This study contributes to picture a review of the Machine Learning and Deep Learning Algorithms used for Autonomous Driving Systems and is organized ba...

X arxiv.org > pdf > 2108.05805.pdf

Reimagining an autonomous vehicle

for driving with machine learning, and grand challenges for research in driving. Keywords: autonomous driving, policy learning, deep learning 1 Introduction The..

¶ news.mit.edu > 2022 > ai-autono...

On the road to cleaner, greener, and faster driving

In a new study, MIT researchers demonstrate a machine-learning approach that can learn to control a fleet of autonomous vehicles as they approach and travel..

S spectrum.ieee.org > how-driveai-i...

How Drive.ai Is Mastering Autonomous Driving With Deep Learning

Among all of the self-driving startups working toward Level 4 autonomy (a self-driving system that doesn't require human intervention in most scenarios), Mounta...

developer.nvidia.com > blog > dee...

End-to-End Deep Learning for Self-Driving Cars

In a new automotive application, we have used convolutional neural networks (CNNs) to map the raw pixels from a front-facing camera to the steering commands..

Choosing learning experience for a machine learning system

SOURCES ↗

geeksforgeeks.org > design-...

Design a Learning System in Machine Learning – GeeksforGeeks

1. Define the Problem: Clearly define the task that the machine learning system is expected to perform. This could be a classification problem (e.g., identifying whether an object in front of the car is a pedestrian or another vehicle), a regression problem (e.g., predicting the trajectory of other vehicles), or a reinforcement learning problem (e.g., learning the optimal driving strategy)[0][2].

2. Select Relevant Data: The choice of training data or experience is crucial as it directly impacts the success or failure of the model. The data should be relevant to the problem and should ideally represent various scenarios that the car might encounter. It could include data from sensors like cameras, lidars, radars, etc., and could cover various driving conditions, traffic scenarios, and geographical locations[0][3].

3. Consider Feedback Mechanism: The learning experience should be able to provide direct or indirect feedback regarding the choices made by the system. This is especially important for reinforcement learning problems where the system learns from the consequences of its actions. For example, penalties could be assigned for violating traffic rules or getting too close to other vehicles[0][2].

4. Ensure Data Quality: The quality of data is critical for the success of the machine learning system. The data should be clean, accurate, and free of biases. In addition, the system should be designed to handle missing or noisy data. Data preprocessing techniques like normalization, outlier detection, and missing value imputation can be used to improve data quality[3][5].

5. Define Performance Measures: Define clear metrics to measure the performance of the machine learning system. These could include machine learning-specific measures like accuracy, precision, recall, etc., as well as business-specific key performance indicators (KPIs) like the ability to avoid accidents, adherence to traffic rules, and passenger comfort[2][5].

6. Consider Computational Resources: The choice of learning experience should also consider the computational resources available. Training a machine learning system on large and complex datasets can be computationally intensive. Therefore, the choice of learning experience should strike a balance between complexity and computational feasibility[2][5].

7. Plan for Long Term: Machine learning is not a "one and done" process. The learning experience should be designed to allow continuous learning and adaptation. This includes updating the model as new data becomes available and retraining the model to adapt to changing conditions[3].

By considering these factors, you can choose a suitable learning experience for a machine learning system in the context of designing an autonomous car[0][2][3][5].

CITATIONS

[0] geeksforgeeks.org

[2] studytonight.com

[3] mitsloan.mit.edu

[5] techtarget.com

How will you choose target function for a machine learning system? (05)

ANSWER | GPT-3.5 MODEL

Choosing the target function for a machine learning system involves determining the specific task or problem that the system is expected to solve. The target function represents the relationship between the

ANSWER | GPT-3.5

Step 1) Choosing the Training Experience: The very important and first task is to choose the training data or training experience which will be fed..

www.coursera.org > articles > ma...

Machine Learning Skills: Your Guide to Getting Started

Technical skills: Software engineering skills The most important concepts in machine learning often involve statistical analysis and mathematical data manipulat...

www.studytonight.com > post > d...

Designing a Learning System: The first step to Machine Learning

Designing a learning system is the crucial first step toward implementing machine learning algorithms effectively. A well-designed learning system lays the foun...

mitsloan.mit.edu > ideas-made-t...

7 lessons to ensure successful machine learning projects

1. Make sure you have easy access to necessary data — and a comprehensive data strategy Successful machine learning solutions start with a strong data..

mitsloan.mit.edu > ideas-made-t...

Machine learning, explained | MIT Sloan

Machine learning is a subfield of artificial intelligence, which is broadly defined as the capability of a machine to imitate intelligent human behavior. Artifi...

1 2

choose target function
machine learning system

SOURCES ↗ ↘

reason.town > target-function-m...

How to Use Target Function in Machine Learning –

input data (features) and the desired output (labels or predictions). Here are the steps to choose the target function:

1. Define the Problem: Clearly define the problem that the machine learning system needs to solve. This could be a classification problem, regression problem, anomaly detection, or any other task. For example, in the context of an autonomous car, the target function could be to classify objects on the road (pedestrians, cars, etc.), predict the trajectory of other vehicles, or detect anomalies in the sensor data[0][2].
2. Determine the Output Format: Based on the problem definition, determine the format of the desired output. It could be a single value (e.g., predicting a continuous variable like speed), a binary value (e.g., classifying objects into two categories), or a multi-class output (e.g., classifying objects into multiple categories)[0][2].
3. Consider the Data Availability: Assess the availability and quality of the data that can be used to train the machine learning system. The target function should be chosen based on the data that is accessible and relevant to the problem. For example, if labeled data is available, supervised learning can be used. If labeled data is limited, semi-supervised or unsupervised learning may be more suitable[2][3].
4. Consider the Evaluation Metrics: Determine the evaluation metrics that will be used to assess the performance of the machine learning system. The choice of target function should align with the evaluation metrics. For example, if accuracy is the primary metric, a classification target function would be appropriate. If mean squared error is the metric, a regression target function would be suitable[2][5].
5. Domain Knowledge and Constraints: Consider any domain-specific knowledge or constraints that need to be incorporated into the target function. For example, in the case of an autonomous car, there may be specific traffic rules or safety considerations that need to be accounted for in the target function[0][2].
6. Iterative Approach: It is often beneficial to start with a simple target function and progressively refine it based on feedback and model performance. This iterative approach allows for fine-tuning and improvement of the target function over time[3].

By following these steps and considering the problem definition, data availability, evaluation metrics, domain knowledge, and an iterative approach, you can choose an appropriate target function for a machine learning system[0][2][3][5].

CITATIONS

- [0] reason.town [2] studytonight.com [3] geeksforgeeks.org
[5] jmvidal.cse.sc.edu

How will you select target function representation for a machine learning system? (05)

ANSWER | GPT-3.5 MODEL

Selecting the target function representation for a machine learning system involves choosing the appropriate model or algorithm that can effectively capture the relationship between the input features and the desired output. Here are the steps to select the target function representation:

Machine Learning –

reason.town

The first step is to choose a supervised learning algorithm. This will be the algorithm that will learn from the data and try to approximate..

🔗 www.techtarget.com > whatis > d...

What is target function? |

Definition from TechTarget

A target function, in machine learning, is a method for solving a problem that an AI algorithm parses its training data to find. Once an..

🔗 www.studytonight.com > post > d...

Designing a Learning System:

The first step to Machine

Learning

Choose Target Function Let's take the example of a checkers-playing program that can generate the legal moves (M) from any board state (B). The program..

🔗 www.geeksforgeeks.org > design-...

Design a Learning System in

Machine Learning –

GeeksforGeeks

Step 2- Choosing target function: The next important step is choosing the target function. It means according to the knowledge fed to the algorithm the..

🔗 machinelearningmastery.com > h...

How Machine Learning

Algorithms Work (they learn a mapping of input to ...

Learning a Function. Machine learning algorithms are described as learning a target function (f) that best maps input variables (X) to an output variable (Y)...

1 2

machine learning target function representation selection

SOURCES ↗ ⟲

🔗 machinelearningmastery.com > fe...

How to Choose a Feature Selection Method For Machine Learning

1. Understand the Problem: Gain a deep understanding of the problem and the nature of the data. Determine whether it is a regression problem, classification problem, anomaly detection, or any other task. This understanding will guide the selection of the target function representation[0][2].
2. Consider the Complexity of the Problem: Assess the complexity of the problem and the underlying patterns in the data. This will help determine the complexity of the target function representation needed. For simple problems, linear models may suffice, while for more complex problems, nonlinear models like decision trees, support vector machines, or neural networks may be necessary[0][2][3].
3. Evaluate Model Types: Explore different types of models that are commonly used for similar problems. Consider the strengths, weaknesses, and assumptions of each model type. For example:
 - Linear models (e.g., linear regression, logistic regression) assume a linear relationship between the input features and the output. These models are simple and interpretable but may not capture complex patterns in the data.
 - Decision trees and random forests can handle nonlinear relationships and interactions between features. They are also capable of feature selection and handling missing values.
 - Support vector machines are effective for binary classification problems and can handle high-dimensional data.
 - Neural networks, particularly deep learning models, are highly flexible and can learn complex patterns but may require large amounts of data and computational resources[0][2][3][5].
4. Consider Model Complexity and Interpretability: Balance the complexity of the model with its interpretability. Simpler models like linear regression or decision trees are more interpretable but may have limited predictive power. More complex models like neural networks may have higher predictive power but are often considered black boxes in terms of interpretability[0][2][3][5].
5. Evaluate Performance and Generalization: Assess the performance of different target function representations using appropriate evaluation metrics and cross-validation techniques. Consider how well each model generalizes to unseen data. Models that perform well on both training and validation/test data are more likely to generalize well to new data[2][3][5].
6. Consider Computational Constraints: Take into account the computational resources available for training and inference. Some models, like deep learning models, may require significant computational power and memory. If computational constraints are a concern, simpler models or model compression techniques can be considered[5][7].

By considering the complexity of the problem, evaluating different model types, assessing model complexity and interpretability, evaluating performance and generalization, and considering computational constraints, you can select an appropriate target function representation for a machine learning system[0][2][3][5][7].

CITATIONS

- [0] machinelearningmastery.com
- [2] stats.stackexchange.com
- [3] reason.town
- [5] machinelearningmastery.com
- [7] studynight.com

1. Feature Selection Methods Feature selection methods are intended to reduce the number of input variables to those that are believed to be most useful..

 jmvidal.cse.sc.edu > talks > mlintr... Choose Representation for Target Function – University of South Carolina

Introduction to Machine Learning. Choose Representation for Target Function. a table? a collection of rules? a neural network ?..

 stats.stackexchange.com > questi... What exactly is a representation in the context of machine learning ...

representation as the way the registers are encoded in order to be used by an algorithm. representation as a pattern or regularity in the data...

 reason.town > target-function-m... How to Use Target Function in Machine Learning – reason.town

In machine learning, the target function is typically a function that maps input data to output labels. For example, in a classification task, the target..

 www.youtube.com > watch?v=Zv... Lecture 4-Choosing a Representation for the Target Function ... – YouTube INDIA Lecture 4-Choosing a Representation for the Target Function Designing a learning systemIntro to ML 261 views Jan 27, 2021 3 Dislike Share Save Programmi..

How will you select algorithm for a machine learning system? (05)

ANSWER | GPT-3.5 MODEL

Selecting the algorithm for a machine learning system involves considering various factors such as the problem type, the available data, the desired performance, and computational constraints. Here are the steps to select an algorithm for a machine learning system:

1. **Problem Type:** Identify the problem type, whether it is a classification, regression, clustering, or another type of problem. Different algorithms are designed to tackle specific problem types, so understanding the problem type is crucial in selecting the appropriate algorithm[0][2].
2. **Data Size and Complexity:** Consider the size and complexity of the available data. Some algorithms perform better with large datasets, while others are more suitable for smaller datasets. Additionally, if the data has a high dimensionality or contains complex relationships, algorithms like deep learning or ensemble methods may be more effective[0][2][3].
3. **Algorithm Performance:** Evaluate the performance of different algorithms on similar problems or benchmark datasets. Consider metrics such as accuracy, precision, recall, F1 score, mean squared error, or other relevant evaluation measures. Look for algorithms that have demonstrated high performance in similar scenarios[0][2][3].
4. **Model Interpretability:** Consider whether interpretability is important for your use case. Some algorithms like decision trees or logistic regression provide interpretable models, making it easier to understand the reasoning behind the predictions. On the other hand, algorithms like neural networks or support vector machines may offer higher performance but are often considered less interpretable[0][2][3].
5. **Computational Resources:** Take into account the computational resources available. Certain algorithms, such as deep learning models, may require significant computational power and memory for training and inference. If computational constraints are a concern, consider algorithms that are computationally efficient or explore model compression techniques[3][5].
6. **Domain Knowledge:** Consider any domain-specific knowledge or constraints that need to be incorporated into the model. Some algorithms may be more suited to handle specific characteristics or requirements of the problem domain. For example, decision trees are often used in healthcare due to their interpretability and ability to handle missing data[0][2].
7. **Iterative Approach:** It is often beneficial to start with simpler algorithms and gradually increase complexity if needed. This iterative approach allows for fine-tuning and improvement of the model over time[3].

By considering the problem type, data size and complexity, algorithm performance, model interpretability, computational resources, domain knowledge, and an iterative approach, you can select an appropriate algorithm for a machine learning system[0][2][3][5].

CITATIONS

[0] learn.microsoft.com

[2] towardsdatascience.com

[3] ibm.com

[5] en.wikipedia.org

select algorithm machine

learning system

SOURCES ↴

 learn.microsoft.com > en-us > azu...

[How to select a machine learning algorithm – Azure Machine Learning ...](#)

The algorithm you select depends primarily on two different aspects of your data science scenario: What you want to do with your data? Specifically, what..

 www.ibm.com > garage > method ...

[Evaluate and select a machine learning algorithm – IBM](#)

Selecting the correct algorithm involves conducting experiments, evaluating machine learning algorithms, and tuning parameters. ← Back to Reason practices Evalu..

 towardsdatascience.com > how-t...

[How to Select the Right Machine Learning Algorithm](#)

For any given machine learning problem, numerous algorithms can be applied and multiple models can be generated. A spam detection classification problem, for ex..

 www.ibm.com > cloud > architect...

[Evaluate and select a machine learning algorithm – IBM](#)

Evaluate and select a machine learning algorithm Different machine learning algorithms search for different trends and patterns. One algorithm isn't the best ac..

 learn.microsoft.com > en-us > azu...

[Machine Learning Algorithm Cheat Sheet – designer – Azure Machine ...](#)

The Azure Machine Learning Algorithm Cheat Sheet helps you choose the right algorithm from the designer for a predictive analytics model. Note Designer supports..

What is overfitting? How will you prevent overfitting?

ANSWER | GPT-3.5 MODEL

Overfitting refers to a situation in machine learning where a model becomes too closely aligned with the training data and fails to generalize well to new, unseen data. It occurs when the model learns the noise or random fluctuations in the training data, rather than the underlying patterns or relationships. As a result, an overfit model may perform well on the training data but gives inaccurate predictions on new data.

To prevent overfitting and improve the generalization ability of a machine learning model, several techniques can be employed:

1. Increase Training Data: One effective way to prevent overfitting is to increase the amount of training data. More data can help the model to capture the underlying patterns and reduce the influence of noise or random fluctuations. Collecting more data or augmenting the existing data can help improve the model's performance[1][5].
2. Feature Selection: Careful feature selection can help prevent overfitting. It involves choosing the most relevant and informative features for the model. Removing irrelevant or redundant features can reduce the complexity of the model and minimize the chances of overfitting. Techniques such as correlation analysis, feature importance, or domain knowledge can aid in feature selection[1][5].
3. Regularization: Regularization is a technique used to prevent overfitting by adding a penalty term to the loss function during model training. This penalty discourages the model from fitting the noise in the data. L1 regularization (Lasso) and L2 regularization (Ridge) are common regularization techniques. They add a regularization term to the loss function, which helps to shrink the model's parameters or coefficients. This, in turn, reduces overfitting and improves generalization[1][5].
4. Cross-Validation: Cross-validation is a technique used to estimate the performance of a model on unseen data. It involves splitting the available data into multiple subsets, training the model on a portion of the data, and evaluating its performance on the remaining portion. By repeating this process with different splits, cross-validation provides a more robust estimate of the model's generalization ability. It helps detect overfitting by assessing how well the model performs on unseen data[1][5].
5. Early Stopping: Early stopping is a technique commonly used in iterative training algorithms, such as gradient descent. It involves monitoring the model's performance on a validation set during training and stopping the training process when the performance starts to deteriorate. This prevents the model from overfitting by finding the point where further training leads to diminishing returns or increased overfitting[1][5].
6. Ensemble Methods: Ensemble methods combine multiple models to make predictions. They can help prevent overfitting by reducing the variance and bias of the individual models. Techniques such as bagging (e.g., random forests) and boosting (e.g., AdaBoost, Gradient Boosting) are commonly used ensemble methods that improve the model's generalization ability[1][5].

By employing techniques such as increasing training data, feature selection, regularization, cross-validation, early stopping, and ensemble methods, you can prevent overfitting and improve the generalization ability of a machine learning model[1][5].

overfitting prevent

overfitting machine

learning

SOURCES ↗

🔗 elitedatascience.com > overfitting...

Overfitting in Machine Learning: What It Is and How to Prevent It

Signal vs. Noise Goodness of Fit
Overfitting vs. Underfitting How to Detect Overfitting in Machine Learning
How to Prevent Overfitting in Machine Learning Addit..

🔗 machinelearningmastery.com > in...

How to Avoid Overfitting in Deep Learning Neural Networks

This tutorial is divided into four parts; they are: The Problem of Model Generalization and Overfitting Reduce Overfitting by Constraining Model Complexity Meth..

🔗 towardsdatascience.com > 8-sim...

8 Simple Techniques to Prevent Overfitting | by David Chuan-En Lin ...

Overfitting is a very common problem in Machine Learning and there has been an extensive range of literature dedicated to studying methods for preventing overfi..

💡 blog.ml.cmu.edu > 2020 > 08 > 31 ...

4 – The Overfitting Iceberg – Machine Learning Blog | ML@CMU

Figure 1. Modern ML practitioners witness phenomena that cast new insight on the bias-variance trade-off philosophy. The evidence that very complex neural netwo..

🔗 www.ibm.com > topics > overfitting

What is Overfitting? | IBM

If overtraining or model complexity results in overfitting, then a logical prevention response would be either to pause training process earlier, also known as,..

CITATIONS

[1] machinelearningmastery.com

[5] aws.amazon.com

What is underfitting? How will you avoid underfitting?

ANSWER | GPT-3.5 MODEL

Underfitting in machine learning occurs when a statistical model or algorithm fails to capture the underlying trend or patterns in the data. It typically happens when the model is too simple or lacks the necessary complexity to accurately represent the data. As a result, an underfit model may perform poorly on both the training data and new, unseen data. Underfitting can be visualized as a model that is too loose and unable to capture the true relationships in the data.

To avoid underfitting and improve the performance of a machine learning model, the following approaches can be taken:

1. Increase Model Complexity: If the model is too simple and fails to capture the underlying patterns in the data, increasing the complexity of the model can help. This can be done by adding more layers or increasing the number of parameters in the model. For example, in neural networks, increasing the number of hidden layers or neurons can enhance the model's ability to capture complex relationships[2][3].
2. Add More Features: If the model lacks the necessary features to represent the data accurately, adding more relevant features can help improve the model's performance. Feature engineering techniques can be used to extract or create additional features from the existing data. By including more informative features, the model can better capture the underlying trends and patterns[2][3].
3. Reduce Regularization: Regularization techniques like L1 or L2 regularization can help prevent overfitting, but they can also lead to underfitting if applied excessively. By reducing the strength of regularization, the model's flexibility can be increased, allowing it to better fit the data. This can be achieved by adjusting the regularization hyperparameters, such as the regularization strength or the penalty term[2][3].
4. Increase Training Data: Insufficient training data can contribute to underfitting. By increasing the amount of training data, the model has a better chance of capturing the underlying trends and patterns. Collecting more data or using techniques like data augmentation can help mitigate the issue of underfitting due to limited data[2][3].
5. Ensemble Methods: Ensemble methods, such as bagging or boosting, can help overcome underfitting by combining multiple models. By training multiple models and aggregating their predictions, ensemble methods can improve the overall performance and capture more complex relationships in the data[3].
6. Evaluate Model Complexity: It is important to strike a balance between model complexity and performance. While increasing the model complexity can help overcome underfitting, it is also important to avoid overfitting. Regularly evaluate the model's performance on both training and validation data to find the optimal level of complexity that achieves the best trade-off between bias and variance[3].

By increasing the model complexity, adding more relevant features,

underfitting avoid
techniques

SOURCES ↗

🔗 www.ibm.com > topics > underfitt...

What is Underfitting? | IBM

There are a number of different methods, such as L1 regularization, Lasso regularization, dropout, etc., which help to reduce the noise and outliers within a..

🔗 blog.pythian.com > deep-learning...

Deep Learning: Techniques to
Avoid Overfitting and
Underfitting

Deep Learning: Techniques to Avoid Overfitting and Underfitting by Manas Narkar May 29, 2020 Posted in:
Technical Track Tags: deep learning, generalization, mac..

🔗 www.geeksforgeeks.org > underfitt...

ML | Underfitting and
Overfitting - GeeksforGeeks

Underfitting can be avoided by using more data and also reducing the features by feature selection. In a nutshell, Underfitting refers to a model that..

🔗 towardsdatascience.com > techni...

Techniques for handling
underfitting and overfitting in
Machine ...

1 Photo by Pietro Jeng on Unsplash I'll be talking about various techniques that can be used to handle overfitting and underfitting in this article...

🔗 allcloud.io > blog > how-to-solve-...

How to Solve Underfitting and
Overfitting Data Models |
AllCloud

There are 4 main techniques you can try: Adding more data Your model is overfitting when it fails to generalize to new data. That means..

reducing regularization, increasing training data, using ensemble methods, and evaluating model complexity, underfitting can be avoided, and the model's performance can be improved[2][3].

CITATIONS

[2] geeksforgeeks.org

[3] towardsdatascience.com

Why do we use regression analysis?

ANSWER | GPT-3.5 MODEL

Regression analysis is used to understand and describe the relationships between a dependent variable and one or more independent variables. It allows us to examine how changes in the independent variables are associated with changes in the dependent variable. Here are some reasons why regression analysis is commonly used:

1. **Relationship Description:** Regression analysis helps in describing and quantifying the relationship between variables. It provides a regression equation where the coefficients represent the relationship between each independent variable and the dependent variable. This equation allows us to understand the direction and magnitude of the effect of independent variables on the dependent variable[0].
 2. **Prediction:** Regression analysis can be used for prediction purposes. Once the relationship between the independent and dependent variables is established, the regression equation can be used to make predictions about the dependent variable based on the values of the independent variables. This is particularly useful when we want to estimate or forecast an outcome based on the available data[0].
 3. **Variable Control:** Regression analysis allows for the control of other variables. By including multiple independent variables in the regression model, we can examine the relationship between the dependent variable and each independent variable while controlling for the effects of other variables. This helps in isolating the specific effect of each independent variable on the dependent variable[0].
 4. **Model Interpretation:** Regression analysis provides interpretable models. The coefficients in the regression equation represent the change in the dependent variable associated with a one-unit change in the corresponding independent variable, holding other variables constant. This allows for the interpretation and understanding of the impact of each independent variable on the dependent variable[0].
 5. **Model Evaluation:** Regression analysis enables the evaluation of the statistical significance and goodness of fit of the model. Statistical tests can be conducted to determine if the relationship between the independent and dependent variables is significant. Additionally, metrics such as R-squared can be used to assess how well the regression model fits the data[7].
- Overall, regression analysis is a powerful statistical tool that allows us to understand, describe, and predict relationships between variables. It provides flexibility in analyzing different types of relationships and controlling for other variables, making it useful in various circumstances[0][7].

CITATIONS

[0] statisticsbyjim.com

[7] corporatefinanceinstitute.com

reasons for using regression analysis

SOURCES ↗ ↘

alpha statisticsbyjim.com > regression >...

When Should I Use Regression Analysis?

Use Regression Analysis to Control the Independent Variables As I mentioned, regression analysis describes how the changes in each independent variable are rela..

triangle www.alchemer.com > resources > ...

What is Regression Analysis and Why Should I Use It?

Regression analysis is a powerful statistical method that allows you to examine the relationship between two or more variables of interest. While there are many..

hbr.org > 2015 > 11 > a-refresher-o...

A Refresher on Regression Analysis

The good news is that you probably don't need to do the number crunching yourself (hallelujah!) but you do need to correctly understand and interpret..

towardsdatascience.com > 3-rea...

3 Reasons Why You Should Use Linear Regression Models Instead of Neural ...

Point #1. Regression analysis is more versatile and has wide applicability. Linear regression and Neural networks are both models that you can use to make..

globe www.g2.com > articles > regressio...

What Is Regression Analysis? Types, Importance, and Benefits

The primary purpose of regression analysis is to describe the relationship between variables, but it can also be used to: Estimate the value of one..

What is covariance? Why and how we use covariance in linear regression?

ANSWER | GPT-3.5 MODEL

Covariance is a statistical measure that quantifies the relationship between two random variables. It measures how changes in one variable are associated with changes in another variable. In the context of linear regression, covariance is used to assess the relationship between the independent variables and the dependent variable. Here's why and how covariance is used in linear regression:

1. Covariance in Linear Regression: In linear regression, the goal is to find the best-fit line that represents the relationship between the independent variables (features) and the dependent variable (target). Covariance is used to estimate the strength and direction of the linear relationship between the independent variables and the dependent variable.
2. Covariance Calculation: Covariance is calculated using the formula: 

Where:

- X and Y are the corresponding data points of the two variables.
- n is the number of data points.

The resulting covariance value can be positive, indicating a positive relationship, or negative, indicating a negative relationship. A covariance of zero indicates no linear relationship between the variables.

3. Using Covariance in Linear Regression: In linear regression, covariance is used to estimate the coefficients (slopes) of the regression equation. The coefficients represent the change in the dependent variable for a one-unit change in the corresponding independent variable. The covariance between each independent variable and the dependent variable is calculated, and then used to estimate the coefficients through the method of least squares.
4. Covariance Matrix: In multiple linear regression, where there are multiple independent variables, the covariance between each pair of independent variables is calculated to form a covariance matrix. This matrix provides valuable information about the relationships and interactions between the independent variables. It helps in assessing multicollinearity, which is the presence of high correlation between independent variables.
5. Assumptions and Interpretation: Linear regression assumes that the independent variables and the residuals (the differences between the actual and predicted values) have a covariance of zero. This assumption is important for the accuracy and reliability of the regression model. Additionally, the sign of the covariance between an independent variable and the dependent variable indicates the direction of their relationship.

In summary, covariance is used in linear regression to estimate the coefficients of the regression equation, assess the strength and direction of the relationship between the independent variables and the dependent variable, and evaluate multicollinearity. It plays a crucial role in understanding and modeling the relationships in linear regression [0][1].

CITATIONS

[0] stats.libretexts.org

[1] stats.stackexchange.com

covariance linear
regression use and
benefits

SOURCES ↗

🔗 stats.libretexts.org > Bookshelves ...

12: Variance, Covariance, and Linear Regression

Paul Pfeiffer Rice University 12.1:
Variance 12.2: Covariance and the
Correlation Coefficient 12.3: Linear
Regression 12.4: Problems on Variance,
Covariance, Li..

🔗 stats.stackexchange.com > questi...

Why is a regression

coefficient

covariance/variance

Perhaps the most straightforward ones rely on the most basic characterizations of the situations: namely, the correlation coefficient is defined as the mean pro..

🔗 stats.libretexts.org > Bookshelves ...

12.2: Covariance and the Correlation Coefficient

Definition: Correlation Coefficient. The correlation coefficient $\rho = \rho[X, Y]$ is the quantity. $\rho[X, Y] = E[X * Y *] = E[(X - \mu_X)(Y - \mu_Y)]$

🔗 en.wikipedia.org > wiki > Analysis..._

Analysis of covariance – Wikipedia

[1] The ANCOVA model assumes a linear relationship between the response (DV) and covariate (CV): In this equation, the DV, is the jth observation under.

🔗 stats.libretexts.org > Bookshelves ...

12.3: Linear Regression – Statistics LibreTexts

Then, (ρ^2) is interpreted as the fraction of uncertainty removed by the linear rule and X. This interpretation should not be pushed too far, but..

What is scatter plot? How scatter plot helps us to select appropriate regression method?

ANSWER | GPT-3.5 MODEL

A scatter plot is a graphical representation of the relationship between two variables. It displays data points as individual dots on a Cartesian plane, with the x-axis representing one variable and the y-axis representing the other variable. Scatter plots are useful for visualizing the distribution, pattern, and potential relationship between the variables. They can help in selecting an appropriate regression method by providing insights into the nature of the relationship between the variables. Here's how scatter plots help in selecting an appropriate regression method:

1. **Visualizing the Relationship:** Scatter plots allow us to visually examine the relationship between the variables. By plotting the data points, we can observe the overall pattern, trend, or lack of a relationship between the variables. Depending on the nature of the relationship observed, different regression methods may be more suitable.
2. **Linearity:** Scatter plots help in assessing the linearity of the relationship between the variables. If the data points form a roughly straight line or follow a clear curve, a linear regression may be appropriate. On the other hand, if the data points do not follow a clear pattern or exhibit a nonlinear relationship, alternative regression methods like polynomial regression, exponential regression, or logistic regression may be more appropriate.
3. **Outliers:** Scatter plots can help in identifying outliers, which are data points that deviate significantly from the overall pattern. Outliers can have a strong influence on the regression analysis and may affect the validity of the results. By visualizing the scatter plot, we can identify and assess the impact of outliers on the relationship between the variables. In some cases, robust regression methods or outlier detection techniques may be necessary to handle outliers appropriately.
4. **Homoscedasticity:** Scatter plots can also help in assessing the homoscedasticity assumption, which assumes that the variability of the dependent variable is constant across all levels of the independent variable(s). If the spread of the data points is approximately equal across the range of the independent variable(s), a simple linear regression may be appropriate. However, if the variability changes systematically with the values of the independent variable(s), other regression methods like weighted least squares or generalized least squares may be more suitable.
5. **Interaction Effects:** Scatter plots can reveal potential interaction effects between the independent variables. An interaction effect occurs when the relationship between the dependent variable and one independent variable depends on the value of another independent variable. By visualizing the scatter plot, we can detect if there are any patterns or changes in the relationship between the variables based on the values of other variables. In such cases, regression methods that account for interaction effects, such as multiple regression or hierarchical regression, may be needed.

In summary, scatter plots provide a visual representation of the relationship between variables, helping to assess linearity, outliers,

scatter plot role in selecting appropriate regression method

SOURCES ↗

openstax.org > books > contempo...

8.8 Scatter Plots, Correlation, and Regression Lines – Contemporary ...

The statistical method of regression can find a formula that does the best job of predicting a score on the final exam based on the..

www.kristakingmath.com > blog > ...

Scatterplots and regression lines

A scatterplot, also called a scattergraph or scatter diagram, is a plot of the data points in a set. It plots data that takes two..

www.khanacademy.org > math > s...

Scatterplots and correlation review (article) | Khan Academy

A scatterplot is a type of data display that shows the relationship between two numerical variables. Each member of the dataset gets plotted as a..

openstax.org > books > introduct...

12.2 Scatter Plots – Introductory Statistics | OpenStax

Using the TI-83, 83+, 84, 84+ Calculator. To create a scatter plot: Enter your X data into list L1 and your Y data into list..

cqeacademy.com > cqe-body-of...

The Scatter Plot & Linear Regression

The Scatter Plot & Linear Regression The Scatter Plot is one of the seven QC Tools that you, the Quality Engineer, must know and be..

homoscedasticity, and interaction effects. By analyzing the scatter plot, we can gain insights that guide us in selecting the appropriate regression method for modeling the relationship between the variables [2][3].

CITATIONS

[2] khanacademy.org

[3] openstax.org

What is sum of squares of residuals? How will you use SSR to calculate parameters for linear regression?

ANSWER | GPT-3.5 MODEL

The sum of squares of residuals (SSR), also known as the sum of squared errors (SSE), is a measure of the overall discrepancy between the observed values and the predicted values in a linear regression model. It quantifies the unexplained variation or the residuals (the differences between the observed and predicted values). In linear regression, SSR is used to calculate the parameters of the regression model through the method of least squares. Here's how SSR is used to calculate the parameters for linear regression:

1. Residual Calculation: In linear regression, the predicted values (\hat{y}) are obtained by applying the regression equation to the independent variable(s) (x). The residuals (e) are then calculated as the differences between the observed values (y) and the predicted values: $e = y - \hat{y}$.
2. Sum of Squares of Residuals: SSR is calculated by squaring each residual and summing them up: $SSR = \sum(e^2)$. This provides a measure of the total unexplained variation in the dependent variable that is not accounted for by the regression model.
3. Minimizing SSR: The goal in linear regression is to find the line that minimizes the SSR. The regression line is defined by the equation: $\hat{y} = \beta_0 + \beta_1 x$, where β_0 is the intercept and β_1 is the slope. The values of β_0 and β_1 are estimated such that the SSR is minimized.
4. Method of Least Squares: The method of least squares is used to estimate the parameters (β_0 and β_1) that minimize SSR. This method finds the values of β_0 and β_1 that minimize the sum of the squared residuals. Mathematically, this is done by taking the partial derivatives of SSR with respect to β_0 and β_1 , setting them to zero, and solving the resulting equations.
5. Normal Equations: The normal equations are derived from the partial derivatives and provide a system of equations that can be solved to find the parameter estimates: 

Where:

- n is the number of data points.
 - Σx and Σy are the sums of the independent and dependent variables, respectively.
 - Σxy and Σx^2 are the sums of the products of the independent and dependent variables, and the squares of the independent variables, respectively.
6. Parameter Estimation: By solving the normal equations, the parameter estimates for the linear regression model can be obtained. The estimated intercept (β_0) and slope (β_1) are used to define the regression line that minimizes SSR and provides the best fit to the data.

In summary, SSR is the sum of the squared residuals and represents the unexplained variation in a linear regression model. By minimizing

sum of squares of residuals linear regression calculation

SOURCES

 www.statisticshowto.com > resid...

Sum of Squares: Residual Sum, Total Sum, Explained Sum, Within

Step 1: Find the mean by adding the numbers together and dividing by the number of items in the set: $(3 + 5 + 7) / 3 = 5$

 www.wallstreetmojo.com > residu...

Residual Sum of Squares – Meaning, Formula, Examples, Calculate

To calculate RSS, first find the model's level of error or residue by subtracting the actual observed values from the estimated values. Then, square and add the results.

 www.investopedia.com > terms > ...

Residual Sum of Squares (RSS): What It Is, How to Calculate It

Adam Barone Updated May 28, 2022

Reviewed by Cierra Murry What is the Residual Sum of Squares (RSS)? The residual sum of squares (RSS) is.

 [www.khanacademy.org](https://www.khanacademy.org/math/s...) > math > s...

Introduction to residuals (article) | Khan Academy

Introduction to residuals Google Classroom Build a basic understanding of what a residual is. We run into a problem in stats when we're trying to...

 www.statology.org > residual-sum...

Residual Sum of Squares Calculator – Statology

This calculator finds the residual sum of squares of a regression equation based on values for a predictor variable and a response variable. Simply enter..

SSR using the method of least squares and solving the normal equations, the parameters (β_0 and β_1) can be estimated to define the regression line that best fits the data[2][3].

CITATIONS

[2] investopedia.com

[3] khanacademy.org

Equation for the Simple Linear Regression Model, Multiple Linear Regression Model and Polynomial Linear Regression Model

ANSWER | GPT-3.5 MODEL

Sure! Here are the equations for the Simple Linear Regression Model, Multiple Linear Regression Model, and Polynomial Linear Regression Model:

1. Simple Linear Regression Model: The simple linear regression model represents the relationship between a single independent variable (x) and a dependent variable (y) with a linear equation:



Where:

- y is the dependent variable.
- β_0 is the intercept, representing the value of y when x is zero.
- β_1 is the slope, representing the change in y for a one-unit change in x .
- ϵ is the error term, representing the random variability or unexplained variation in y .

2. Multiple Linear Regression Model: The multiple linear regression model represents the relationship between multiple independent variables (x_1, x_2, \dots, x_n) and a dependent variable (y) with a linear equation:



Where:

- y is the dependent variable.
- β_0 is the intercept, representing the value of y when all independent variables are zero.
- $\beta_1, \beta_2, \dots, \beta_n$ are the slopes, representing the change in y for a one-unit change in each respective independent variable.
- ϵ is the error term, representing the random variability or unexplained variation in y .

3. Polynomial Linear Regression Model: The polynomial linear regression model represents a non-linear relationship between an independent variable (x) and a dependent variable (y) by including higher-order terms in the equation. The general form of the equation is:



Where:

- y is the dependent variable.
- $\beta_0, \beta_1, \beta_2, \dots, \beta_n$ are the coefficients, representing the weights assigned to each term.
- x is the independent variable.
- n is the degree of the polynomial, determining the number of terms in the equation.

It's important to note that these equations provide a general representation of the linear regression models. The specific parameter estimates and equation coefficients are obtained through the process of fitting the regression model to the data using

Equation for Simple Linear Regression Model, Multiple Linear Regression Model, Polynomial Linear Regression Model.

SOURCES

online.stat.psu.edu > stat501 > les...
5.3 – The Multiple Linear Regression Model | STAT 501 – Statistics Online

In the formula, n = sample size, p = number of β parameters in the model (including the intercept) and SSE = sum of squared..

www.scribbr.com > statistics > si...
Simple Linear Regression | An Easy Introduction & Examples – Scribbr

The relationship between the independent and dependent variable is linear: the line of best fit through the data points is a straight line (rather than..)

www.scribbr.com > statistics > m...
Multiple Linear Regression | A Quick Guide (Examples) – Scribbr

The formula for a multiple linear regression is: = the predicted value of the dependent variable..

thomasttam.medium.com > simpl...
Simple and Multiple Linear Regression for Beginners – Medium

Let's break down the formula: Y is the dependent variable we're predictor for X is the independent variable $B1$ is the slope, which determines the..

medium.com > swlh > linear-regre...
Linear Regression (Simple, Multiple and Polynomial) – Medium

$Y_{hat}=a+bX$. A sample graph representing the relation between an independent and dependent variable. In Data Science, Linear regression is one of the most c..

techniques such as the method of least squares[4][5].

CITATIONS

[4] medium.com

[5] online.stat.psu.edu

Define the terms – Bias, variance, underfitting and overfitting

ANSWER | GPT-3.5 MODEL

Bias refers to the error introduced by approximating a real-world problem with a simplified model. In the context of machine learning, bias measures how far off the predictions of a model are from the true values. It represents the model's tendency to consistently underpredict or overpredict the target variable. Here's a breakdown of the term:

- **High Bias:** When a model has high bias, it means that it oversimplifies the underlying relationship between the features and the target variable. A high bias model tends to have low complexity and may not be able to capture the nuances or patterns in the data. This leads to underfitting, where the model performs poorly on both the training and testing data.
- **Low Bias:** Conversely, a model with low bias is more flexible and can capture complex relationships between the features and the target variable. It has a higher capacity to learn from the data and adapt to different patterns. A low bias model is more likely to fit the training data well, but it may also be prone to overfitting.

In summary, bias refers to the difference between the predictions made by a model and the true values. High bias indicates an oversimplified model that underfits the data, while low bias suggests a more complex model that has the potential to overfit the data.

[towardsdatascience.com](#)

Calculate precision and recall for a given class of confusion matrix.

ANSWER | GPT-3.5 MODEL

Define terms Bias variance underfitting overfitting

SOURCES ↗ ↘

⊕ machinelearningcompass.com > ...

[Bias, Variance, and Overfitting Explained, Step by Step](#)

The Tradeoff? An All-Purpose Remedy? Diving Deeper Further Reading Model Optimization Bias, Variance, and Overfitting Explained, Step by Step You have likely he..

⊕ towardsdatascience.com > overfit...

[Overfitting, underfitting, and the bias-variance tradeoff](#)

Overfitting, underfitting, and the bias-variance tradeoff are foundational concepts in machine learning. A model is overfit if performance on the training data...

⊕ towardsdatascience.com > bias-v...

[Bias, Variance and How they are related to Underfitting, Overfitting ...](#)

Member-only story Bias, Variance and How they are related to Underfitting, Overfitting I came across the terms bias, variance, underfitting and overfitting whil..

⊕ towardsdatascience.com > clearl...

[Clearly Explained: What is Bias-Variance tradeoff, Overfitting ...](#)

Bias error is the difference between the predicted data points and the actual data points which was caused because our model was oversimplified. A model..

⊕ en.wikipedia.org > wiki > Overfitting

[Overfitting – Wikipedia](#)

In mathematical modeling, overfitting is "the production of an analysis that corresponds too closely or exactly to a particular set of data, and may therefore..

calculate precision recall given class confusion matrix

SOURCES ↗ ↘

To calculate precision and recall for a given class in a confusion matrix, you need to understand the components of the confusion matrix and the formulas for precision and recall.

A confusion matrix is a table that summarizes the performance of a classification model by displaying the counts of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) for each class.

Precision and recall are two important metrics used to evaluate the performance of a classification model:

- Precision: Precision is the ratio of true positives to the sum of true positives and false positives. It measures the accuracy of positive predictions made by the model. The formula for precision is:

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP})$$

- Recall: Recall, also known as sensitivity or true positive rate, is the ratio of true positives to the sum of true positives and false negatives. It measures the model's ability to correctly identify positive instances. The formula for recall is:

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN})$$

To calculate precision and recall for a specific class in the confusion matrix, you need to identify the corresponding TP, FP, and FN values. Once you have those values, you can apply the formulas mentioned above to calculate precision and recall.

Here's an example to illustrate the calculation:

Suppose we have a binary classification problem with two classes (class A and class B) and the following confusion matrix:

Predicted A		Predicted B
Actual A	50	10
Actual B	5	80

To calculate precision and recall for class A:

- TP (True Positives) = 50
- FP (False Positives) = 5
- FN (False Negatives) = 10

Using the formulas mentioned earlier:

- Precision for class A = $\text{TP} / (\text{TP} + \text{FP}) = 50 / (50 + 5) = 0.909$
- Recall for class A = $\text{TP} / (\text{TP} + \text{FN}) = 50 / (50 + 10) = 0.833$

So, the precision for class A is 0.909 and the recall for class A is 0.833.

Remember that precision and recall are class-specific metrics, and you can calculate them for each class in a multi-class classification problem by considering that class as the positive class and the rest as the negative class.

machinelearningmastery.com

Calculate precision and recall for a given class of confusion matrix.

ANSWER | GPT-3.5 MODEL

To calculate precision and recall for a given class in a confusion matrix, you need to identify the true positives (TP), false positives (FP), and false negatives (FN) for that specific class. Once you have these values, you can apply the formulas for precision and recall.

 stackoverflow.com > questions > ...

[calculate precision and recall in a confusion matrix](#)

`true_pos = np.diag(cm)`
`precision = np.sum(true_pos) / np.sum(cm, axis=0)`
`recall = np.sum(true_pos) / np.sum(cm, axis=1)`

Share Improve this answer Follow...

 machinelearningmastery.com > pr...

[How to Calculate Precision, Recall, and F-Measure for Imbalanced ...](#)

In this tutorial, you will discover how to calculate and develop an intuition for precision and recall for imbalanced classification. After completing this tuto...

 www.bmc.com > blogs > confusio...

[Precision, Recall & Confusion Matrices in Machine Learning](#)

Confusion matrix. Both precision and recall can be interpreted from the confusion matrix, so we start there. The confusion matrix is used to display how...

 stats.stackexchange.com > questi...

[machine learning – How do you calculate precision and recall for ...](#)

How do you calculate precision and recall for multiclass classification using confusion matrix? Ask Question Asked 10 years, 5 months ago Modified 9 months ago..

 towardsdatascience.com > perfor...

[Performance Metrics: Confusion matrix, Precision, Recall, and F1 Score ...](#)

The confusion matrix, precision, recall, and F1 score gives better intuition of prediction results as compared to accuracy. To understand the concepts, we will ..

1 2

calculate precision and recall for specific class confusion matrix

SOURCES

 machinelearningmastery.com > pr...

[How to Calculate Precision, Recall, and F-Measure for Imbalanced ...](#)

Here's how you can calculate precision and recall:

1. Precision: Precision measures the accuracy of positive predictions made by the model. It is calculated as the ratio of true positives to the sum of true positives and false positives.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

2. Recall: Recall, also known as sensitivity or true positive rate, measures the model's ability to correctly identify positive instances. It is calculated as the ratio of true positives to the sum of true positives and false negatives.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

To calculate precision and recall for a given class, follow these steps:

1. Identify the class for which you want to calculate precision and recall.
2. Find the corresponding values in the confusion matrix:
 - TP: The count of true positives for the given class.
 - FP: The count of false positives for the given class.
 - FN: The count of false negatives for the given class.
3. Apply the formulas for precision and recall using the identified values:
 - Precision = $\frac{\text{TP}}{\text{TP} + \text{FP}}$
 - Recall = $\frac{\text{TP}}{\text{TP} + \text{FN}}$

Here's an example to illustrate the calculation:

Suppose we have a binary classification problem with two classes (class A and class B) and the following confusion matrix:

Predicted A Predicted B

Actual A	50	10
Actual B	5	80

To calculate precision and recall for class A:

- TP (True Positives) = 50
- FP (False Positives) = 5
- FN (False Negatives) = 10

Using the formulas mentioned earlier:

- Precision for class A = $\frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{50}{50 + 5} = 0.909$
- Recall for class A = $\frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{50}{50 + 10} = 0.833$

So, the precision for class A is 0.909 and the recall for class A is 0.833.

Remember that precision and recall are class-specific metrics, and you can calculate them for each class in a multi-class classification problem by considering that class as the positive class and the rest as the negative class.

stats.stackexchange.com

In this tutorial, you will discover how to calculate and develop an intuition for precision and recall for imbalanced classification. After completing this tuto...

 stats.stackexchange.com > questi...

[How to calculate precision and recall in a 3 x 3 confusion matrix](#)

So you can calculate precision and recall for each of your classes. For multi-class confusion tables, that's the diagonal elements divided by their row and...

 www.bmc.com > blogs > confusio...

[Precision, Recall & Confusion Matrices in Machine Learning](#)

Both precision and recall can be interpreted from the confusion matrix, so we start there. The confusion matrix is used to display how well a...

 stackoverflow.com > questions > ...

[Scikit-learn, get accuracy scores for each class - Stack Overflow](#)

Share Follow answered Dec 17, 2018 at 22:51 javac 2,819 1 20 22 2 To elaborate, assuming column (hence this answer's "axis=1") represents the actual...

 stats.stackexchange.com > questi...

[machine learning - How do you calculate precision and recall for ...](#)

daiyue 1,281 2 9 7 This docx, Evaluating a classification model – What does precision and recall tell me?, from Compumine provides a simple introduction..

1 2



Ask a followup question

▼ Q