

(19)日本国特許庁(JP)

(12)公開特許公報(A)

(11)公開番号

特開2023-58454
(P2023-58454A)

(43)公開日 令和5年4月25日(2023.4.25)

(51)国際特許分類

G 0 6 N 20/00 (2019.01)

F I

G 0 6 N 20/00

審査請求 未請求 請求項の数 20 O L (全35頁)

(21)出願番号 特願2022-163230(P2022-163230)
 (22)出願日 令和4年10月11日(2022.10.11)
 (31)優先権主張番号 17/450,694
 (32)優先日 令和3年10月13日(2021.10.13)
 (33)優先権主張国・地域又は機関
 米国(US)

(71)出願人 390009531
 インターナショナル・ビジネス・マシー
 ンズ・コーポレーション
 INTERNATIONAL BUSI
 NESS MACHINES CORPO
 RATION
 アメリカ合衆国10504 ニューヨー
 ク州 アーモンク ニュー オーチャード
 ロード
 New Orchard Road, A
 rmonk, New York 105
 04, United States of
 America
 (74)代理人 100112690
 弁理士 太佐 種一

最終頁に続く

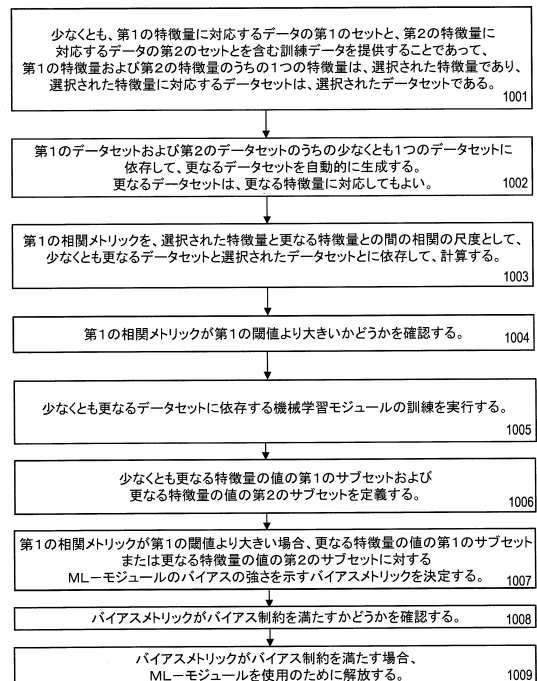
(54)【発明の名称】 機械学習モジュールのバックグラウンドのバイアスの制御のためのコンピュータ実装方法、コンピュータプログラム製品およびコンピュータシステム(機械学習モジュールのバツ

(57)【要約】 (修正有)

【課題】機械学習モジュールを生成するシステム、方法及びプログラムを提供する。

【解決手段】方法は、第1の特徴量に対応する第1のデータセットと第2の特徴量に対応する第2のデータセットを含む訓練データを提供することと、第1のデータセットおよび第2のデータセットに依存する更なるデータセットを生成することと、更なるデータセットおよび選択されたデータセットに依存する第1の相関メトリックを計算することと、第1の相関メトリックが第1の閾値より大きいかどうかを確認することと、更なるデータセットに依存するMLモジュールを訓練することと、更なる特徴量の値の第1のサブセットと更なる特徴量の値の第2のサブセットとを定義することと、第1のサブセットまたは第2のサブセットに対するMLモジュールのバイアスメトリックを決定することと、バイアス制約を確認することと、MLモジュールを解放することと、を含む。

【選択図】図9



【特許請求の範囲】**【請求項 1】**

コンピュータ実装方法であって、前記方法は、

少なくとも第 1 の特徴量に対応する第 1 のデータセットと第 2 の特徴量に対応する第 2 のデータセットとを含む訓練データを提供することであって、前記第 1 の特徴量および前記第 2 の特徴量のうちの 1 つの特徴量は選択された特徴量であり、前記選択された特徴量に対応する前記データセットは選択されたデータセットである、提供することと、

前記第 1 のデータセットおよび前記第 2 のデータセットのうち少なくとも 1 つのデータセットに依存する、更なるデータセットを自動的に生成することであって、前記更なるデータセットは、更なる特徴量に対応する、生成することと、

少なくとも前記更なるデータセットおよび前記選択されたデータセットに依存する、前記選択された特徴量および前記更なる特徴量の間の相関の尺度として、第 1 の相関メトリックを計算することと、

前記第 1 の相関メトリックが第 1 の閾値より大きいかどうかを確認することと、

少なくとも前記更なるデータセットに依存する ML - モジュールの訓練を実行することと、

少なくとも前記更なる特徴量の値の第 1 のサブセットと前記更なる特徴量の値の第 2 のサブセットとを定義することと、

前記第 1 の相関メトリックが前記第 1 の閾値よりも大きい場合、前記更なる特徴量の値の前記第 1 のサブセットまたは前記更なる特徴量の値の前記第 2 のサブセットに対する前記 ML - モジュールのバイアスの強さを示すバイアスメトリックを決定することと、

前記バイアスメトリックがバイアス制約を満たすかどうかを確認することと、

前記バイアスメトリックが前記バイアス制約を満たす場合、前記 ML - モジュールを使用のために解放することと、

を含む、方法。

【請求項 2】

前記選択されたデータセットに依存する前記更なるデータセットを生成することをさらに含む、請求項 1 に記載の方法。

【請求項 3】

前記選択されたデータセットでない前記第 1 のデータセットおよび前記第 2 のデータセットの前記データセットに依存する前記更なるデータセットを生成することをさらに含む、請求項 1 に記載の方法。

【請求項 4】

前記選択されたデータセットと、前記選択されたデータセットでない前記第 1 のデータセットおよび前記第 2 のデータセットの前記データセットと、に依存する前記更なるデータセットを生成することをさらに含む、請求項 1 に記載の方法。

【請求項 5】

前記選択された特徴量はユーザによって選択され、前記方法は、

前記第 1 の特徴量および前記第 2 の特徴量のうち選択された特徴量でない特徴量（非選択特徴量）と、前記選択されたデータセットでない前記第 1 のデータセットおよび前記第 2 のデータセットと前記選択されたデータセットの前記データセットに依存する前記選択された特徴量と、の間の相関の尺度として、第 2 の相関メトリックを計算することと、

前記第 2 の相関メトリックが第 2 の閾値より大きいかどうかを確認することと、

をさらに含む、請求項 1 に記載の方法。

【請求項 6】

前記方法は、前記第 1 の相関メトリックの前記計算、もしくは、前記第 1 の相関メトリックがユーザ相関入力データに依存する前記第 1 の閾値より大きいかどうかの前記確認、またはその両方を実行することをさらに含む、請求項 1 に記載の方法。

【請求項 7】

前記方法は、少なくとも前記更なる特徴量の値の前記第 1 のサブセットとユーザバイア

10

20

30

40

50

ス入力データに依存する前記更なる特徴量の値の前記第 2 のサブセットとを定義することをさらに含む、請求項 1 に記載の方法。

【請求項 8】

前記方法は、それぞれ少なくとも前記更なる特徴量に対応する値を含むアプリケーション入力データセットに依存する前記 ML - モジュールの出力の値を計算するために前記訓練済み ML - モジュールを使用するとき、前記更なる特徴量に関する前記訓練済み ML - モジュールの前記バイアスを監視するための前記バイアスメトリックを前記訓練済み ML - モジュールのメタデータの形態で記憶することをさらに含む、請求項 1 に記載の方法。

【請求項 9】

前記方法は、それぞれ少なくとも前記更なる特徴量に対応する値を含むアプリケーション入力データセットに依存する前記 ML - モジュールの出力の値を計算するために前記訓練済み ML - モジュールを使用するとき、前記更なる特徴量に関する前記訓練済み ML - モジュールの前記バイアスを監視するための前記第 1 の相関メトリックを前記訓練済み ML - モジュールのメタデータの形態で記憶することをさらに含む、請求項 1 に記載の方法。

10

【請求項 10】

前記方法は、それぞれ少なくとも前記非選択特徴量に対応する値を含むアプリケーション入力データセットに依存する前記 ML - モジュールの出力の値を計算するために前記訓練済み ML - モジュールを使用するとき、前記非選択特徴量に関する前記訓練済み ML - モジュールの前記バイアスを監視するための前記更なるバイアスメトリックを前記訓練済み ML - モジュールのメタデータの形式で記憶することをさらに含む、請求項 5 に記載の方法。

20

【請求項 11】

前記方法は、それぞれ少なくとも前記非選択特徴量に対応する値を含むアプリケーション入力データセットに依存する前記 ML - モジュールの出力の値を計算するために前記訓練済み ML - モジュールを使用するとき、前記非選択特徴量に関する前記訓練済み ML - モジュールの前記バイアスを監視するための前記第 2 の相関メトリックを前記訓練済み ML - モジュールのメタデータの形式で記憶することをさらに含む、請求項 5 に記載の方法。

【請求項 12】

前記方法は、
 前記 ML - モジュールの構造を修正することと、
 前記 ML - モジュールの前記訓練の前記実行を繰り返すことと、
 前記バイアスメトリックの前記計算を繰り返すことと、
 前記バイアスメトリックが減少したかどうかを確認することと、
 前記バイアスメトリックが減少していた場合、前記修正された構造を持つ前記 ML - モジュールを使用のために解放すること、および、以前のバージョンの前記 ML - モジュールを拒否することと、
 を含む、請求項 1 に記載の方法。

30

【請求項 13】

前記方法は、
 前記第 1 のデータセットおよび前記第 2 のデータセットの少なくとも 1 つのデータセットに依存する前記更なるデータセットの前記生成を、異なる数学的関数を用いて繰り返すことと、
 前記 ML - モジュールの前記訓練の前記実行を繰り返すことと、
 前記バイアスメトリックの前記計算を繰り返すことと、
 前記バイアスメトリックが減少したかどうかを確認することと、
 前記バイアスメトリックが減少していた場合、前記修正された構造を持つ前記 ML - モジュールを使用のために解放し、以前のバージョンの前記 ML - モジュールを拒否することと、
 を含む、請求項 1 に記載の方法。

40

50

【請求項 14】

前記方法は、前記ML-モジュールの前記構造の前記修正、前記ML-モジュールの前記訓練の前記実行、および前記バイアスメトリックの前記計算を繰り返すことと、前記更なるML-モジュールのバイアスを低減させることについて、更なるML-モジュールの構造の最適化のためのデータベースを提供するために、各繰り返しについて前記ML-モジュールの前記構造を指定するおよび構造パラメータの値のセットを前記それぞれのバイアスメトリックとともに記憶する前記構造パラメータの値のそれぞれの前記セットを記録することと、をさらに含む、

請求項12に記載の方法。

【請求項 15】

前記方法は、前記構造パラメータの値の前記セットを、前記それぞれのバイアスメトリックと共に、前記ML-モジュールのメタデータの形態で記憶することをさらに含む、

請求項14に記載の方法。

【請求項 16】

コンピュータプログラム製品であって、前記コンピュータプログラム製品は、1つまたは複数のコンピュータ可読記憶媒体と、前記1つまたは複数のコンピュータ可読記憶媒体に集合的に記憶されたプログラム命令とを含み、前記プログラム命令は、

少なくとも第1の特徴量に対応する第1のデータセットと第2の特徴量に対応する第2のデータセットとを含む訓練データを提供することであって、前記第1の特徴量および前記第2の特徴量のうちの1つの特徴量は選択された特徴量であり、前記選択された特徴量に対応する前記データセットは選択されたデータセットである、提供することと、

前記第1のデータセットおよび前記第2のデータセットのうち少なくとも1つのデータセットに依存する、更なるデータセットを自動的に生成することであって、前記更なるデータセットは、更なる特徴量に対応する、生成することと、

少なくとも前記更なるデータセットおよび前記選択されたデータセットに依存する、前記選択された特徴量および前記更なる特徴量の間の相関の尺度として、第1の相関メトリックを計算することと、

前記第1の相関メトリックが第1の閾値より大きいかどうかを確認することと、

少なくとも前記更なるデータセットに依存するML-モジュールの訓練を実行することと、

少なくとも前記更なる特徴量の値の第1のサブセットと前記更なる特徴量の値の第2のサブセットとを定義することと、

前記第1の相関メトリックが前記第1の閾値よりも大きい場合、前記更なる特徴量の値の前記第1のサブセットまたは前記更なる特徴量の値の前記第2のサブセットに対する前記ML-モジュールのバイアスの強さを示すバイアスメトリックを決定することと、

前記バイアスメトリックがバイアス制約を満たすかどうかを確認することと、

前記バイアスメトリックが前記バイアス制約を満たす場合、前記ML-モジュールを使用のために解放することと、

を含む、コンピュータプログラム製品。

【請求項 17】

コンピュータシステムであって、前記コンピュータシステムは、

少なくとも第1の特徴量に対応する第1のデータセットと第2の特徴量に対応する第2のデータセットとを含む訓練データを提供することであって、前記第1の特徴量および前記第2の特徴量のうちの1つの特徴量は選択された特徴量であり、前記選択された特徴量に対応する前記データセットは選択されたデータセットである、提供することと、

前記第1のデータセットおよび前記第2のデータセットのうち少なくとも1つのデータセットに依存する、更なるデータセットを自動的に生成することであって、前記更なるデータセットは、更なる特徴量に対応する、生成することと、

少なくとも前記更なるデータセットおよび前記選択されたデータセットに依存する、前記選択された特徴量および前記更なる特徴量の間の相関の尺度として、第1の相関メトリ

10

20

30

40

50

ックを計算することと、

前記第 1 の相関メトリックが第 1 の閾値より大きいかどうかを確認することと、

少なくとも前記更なるデータセットに依存する ML - モジュールの訓練を実行することと、

少なくとも前記更なる特徴量の値の第 1 のサブセットと前記更なる特徴量の値の第 2 のサブセットとを定義することと、

前記第 1 の相関メトリックが前記第 1 の閾値よりも大きい場合、前記更なる特徴量の値の前記第 1 のサブセットまたは前記更なる特徴量の値の前記第 2 のサブセットに対する前記 ML - モジュールのバイアスの強さを示すバイアスメトリックを決定することと、

前記バイアスメトリックがバイアス制約を満たすかどうかを確認することと、

前記バイアスメトリックが前記バイアス制約を満たす場合、前記 ML - モジュールを使用のために解放することと、

を含む、コンピュータシステム。

【請求項 18】

前記選択されたデータセットに依存する更なるデータセットを生成することをさらに含む、請求項 17 に記載のコンピュータシステム。

【請求項 19】

前記選択されたデータセットでない前記第 1 のデータセットおよび前記第 2 のデータセットの前記データセットに依存する前記更なるデータセットを生成することをさらに含む、請求項 17 に記載のコンピュータシステム。

【請求項 20】

前記選択されたデータセットと、前記選択されたデータセットでない前記第 1 のデータセットおよび前記第 2 のデータセットの前記データセットと、に依存する前記更なるデータセットを生成することをさらに含む、請求項 17 に記載のコンピュータシステム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、一般に、機械学習モジュールの分野に関し、特に、機械学習モジュールを生成するための方法に関する。

【背景技術】

【0002】

機械学習モジュールは、様々な分野で応用されている。一般に、機械学習モジュールは、入力値に依存する出力値を計算するために使用され、出力値は、入力値に依存する予測値と見なされることがある。各入力値は、機械学習モジュール (ML - モジュール) のそれぞれの入力特徴量に対応することができる。選択された入力特徴量に対応する値のセットは、サブセットに分割されてもよい。ML - モジュールの特定のアプリケーションに関して、例示的な ML - モジュールの出力値が、選択された入力特徴量に対応する入力値がどのサブセットに属するかに依存しないことが望ましい。言い換えれば、例示的な ML - モジュールは、選択された特徴量に関して、サブセットのうちの 1 つに偏るべきではない。サブセットへの分割は、例示的な ML - モジュールのユーザによって実行されてもよい。

【発明の概要】

【発明が解決しようとする課題】

【0003】

機械学習モジュールのバックグラウンドのバイアスの制御のためのコンピュータ実装方法、コンピュータプログラム製品およびコンピュータシステムを提供する。

【課題を解決するための手段】

【0004】

様々な実施形態は、独立請求項の主題によって説明されるような機械学習モジュール (ML - モジュール) を生成するためのコンピュータシステム、および方法を提供する。有

10

20

30

40

50

利な実施形態は、従属請求項に記載されている。本発明の実施形態は、相互に排他的でない場合、自由に組み合わせることができる。

【0005】

一態様において、本発明は、機械学習モジュール（ML - モジュール）を生成するためのコンピュータ実装方法であって、少なくとも第1の特徴量に対応する第1のデータセットと第2の特徴量に対応する第2のデータセットとを含む訓練データを提供することであって、第1の特徴量および第2の特徴量のうちの1つの特徴量は選択された特徴量であり、選択された特徴量に対応するデータセットは選択されたデータセットである。第1のデータセットおよび第2のデータセットのうち少なくとも1つのデータセットに依存する、更なるデータセットが自動的に生成される。第1の相関メトリックは、少なくとも更なるデータセットおよび選択されたデータセットに依存する、選択された特徴量および更なる特徴量の間の特徴量の間の相関の尺度として、計算される。第1の相関メトリックが第1の閾値より大きいかどうかを確認される。少なくとも更なるデータセットに依存するML - モジュールの訓練が実行される。少なくとも更なる特徴量の値の第1のサブセットと更なる特徴量の値の第2のサブセットとが定義される。第1の相関メトリックが第1の閾値よりも大きい場合、更なる特徴量の値の第1のサブセットまたは更なる特徴量の値の第2のサブセットに対するML - モジュールのバイアスの強さを示すバイアスメトリックが決定される。バイアスメトリックがバイアス制約を満たすかどうかを確認される。バイアスメトリックがバイアス制約を満たす場合、ML - モジュールが使用のために解放される。

10

【0006】

実施形態は、上記のコンピュータ実装方法と実質的に同じ特徴を有するコンピュータシステムおよびコンピュータプログラム製品にさらに向けられる。

20

【0007】

以下の実施形態では、本発明の実施形態について、図面を参照しながら、例示だけのために、より詳細に説明する。

【図面の簡単な説明】

【0008】

【図1】本主題の一例によるコンピュータシステムを模式的に示すブロック図である。

【図2】図1に示すコンピュータシステムとネットワークとの接続を説明するためのブロック図である。

30

【図3】図1に示したコンピュータシステムのML - モジュールを訓練するための訓練レコードを含む訓練データを示す図である。

【図4】図1に示したコンピュータシステムのML - モジュールを示す図である。

【図5】2つのサブセットに分割された更なる特徴量の値を示す。

【図6】図4に示したML - モジュールのバイアスを計算するためのテストデータセットを示す図である。

【図7】図4に示した訓練済みML - モジュールと図6に示したテストデータセットを示す図である。

【図8】図3に示した訓練データを2つのサブセットに分割した非選択特徴量の値を示す図である。

40

【図9】図4に示したML - モジュールを生成するための方法のステップを示す図である。

【発明を実施するための形態】

【0009】

本発明の様々な実施形態の説明は、説明のために提示されるが、開示された実施形態を網羅することまたは限定することを意図するものではない。説明された実施形態の範囲および精神から逸脱することなく、多くの修正および変形が当業者には明らかであろう。本明細書で使用される用語は、実施形態の原理、市場で見られる技術に対する実際の適用または技術的改善を説明するため、または当業者が本明細書に開示される実施形態を理解できるようにするために選択された。

50

【 0 0 1 0 】

更なる特徴量の値の第 1 のサブセットと更なる特徴量の値の第 2 のサブセットとは、不連続なサブセットであってもよい。

【 0 0 1 1 】

訓練データは、各データレコードで構成されてもよく、各データレコードは、第 1 のデータセットのそれぞれの値と第 2 のデータセットのそれぞれの値とを含む。更なるデータセットのそれぞれの値は、それぞれのデータレコードに追加されてもよい。更なるデータセットの生成は、それぞれのデータレコードの第 1 のデータセットの値もしくは第 2 のデータセットの値またはその両方に依存する更なるデータセットのそれぞれの値を生成することを含んでもよい。更なるデータセットのそれぞれの値は、それぞれのデータレコード

10

【 0 0 1 2 】

ML - モジュールの訓練は、ML - モジュールを用いて、少なくともそれぞれのデータレコードの更なるデータセットの値に依存する各データレコードのための ML - モジュールの出力のそれぞれの訓練出力値を計算することを含んでもよい。さらに、ML - モジュールの訓練は、ML - モジュールの出力の訓練出力値に依存する ML - モジュールのモデルパラメータの値を修正することを含んでもよい。

【 0 0 1 3 】

多くの場合、各データレコードは、ターゲット特徴量のそれぞれの値を含んでいてもよい。各データレコードは、それぞれの観測可能なインスタントまたはエンティティに対応してもよい。それぞれの観測可能なインスタントまたはエンティティは、第 1 のデータセットのそれぞれの値および第 2 のデータセットのそれぞれの値、もしくは、それぞれのデータレコードのターゲット特徴量のそれぞれの値、またはその両方によって記述されてもよい。

20

【 0 0 1 4 】

例えば、それぞれの観測可能なインスタントは、第 1 のデータセットのそれぞれの値と第 2 のデータセットのそれぞれの値、もしくはターゲット特徴量のそれぞれの値、またはその両方が、それぞれのデータレコードに書き込まれるまたはキャプチャされるそれぞれの時間間隔、またはそれぞれの時点と関連付けられることがある。言い換えれば、第 1 のデータセットのそれぞれの値と第 2 のデータセットのそれぞれの値、もしくはターゲット特徴量のそれぞれの値、またはその両方は、それぞれ、それぞれの観測可能なインスタントに対応してもよい。それぞれの時間間隔または時点は、機械または通信システムが動作しているそれぞれの時間間隔または時点であってもよい。したがって、データレコードは、時間間隔または時点を含む時間の経過の間の機械または通信システムの動作を記述してもよい。

30

【 0 0 1 5 】

例えば、訓練データは、機械のセンサデータをキャプチャすることによって生成されてもよい。例えば、第 1 のデータセットを記録するために第 1 のセンサが機械に配置されてもよく、第 2 のデータセットを記録するために第 2 のセンサが機械に配置されてもよい。したがって、第 1 のデータセットは、第 1 のセンサ値であってもよく、第 2 のデータセットは、第 2 のセンサ値であってもよい。ターゲット特徴量のそれぞれの値は、機械に配置されてもよい更なるセンサによってキャプチャされてもよい。測定ユニットは、第 1 のデータセットのそれぞれの値、第 2 のデータセットのそれぞれの値、およびターゲット特徴量のそれぞれの値を、それぞれのデータレコードに書き込んでもよい。

40

【 0 0 1 6 】

別の例では、訓練データは、通信システムによって第 1 のデータセット、第 2 のデータセット、もしくはターゲット特徴量の値、またはその組み合わせをキャプチャすることによって提供されてもよい。通信システムは、携帯電話やデスクトップコンピュータなどの 1 つまたは複数の通信装置を含んでいてもよい。通信装置は、インターネットを介して接続され、通信システムを形成してもよい。第 1 および第 2 のデータセットの値、もしくは

50

ターゲット特徴量の値、またはその両方は、それぞれ、1つまたは複数の通信装置によって受信される電子信号の形態で設計されてもよい。

【0017】

一例では、それぞれのデータレコードの第1の特徴量と第2の特徴量、もしくはターゲット特徴量、またはその両方のそれぞれの値は、それぞれのデータレコードに対応するそれぞれの時点またはそれぞれの時間間隔内の通信装置の1つのそれぞれのステータスまたはそれぞれのプロパティ、例えばジオロケーションまたは信号受信の強さを記述してもよい。

【0018】

更なる例によれば、それぞれのデータレコードの第1の特徴量および第2の特徴量、もしくはターゲット特徴量、またはその両方の値は、それぞれ、それぞれのエンティティ、例えば通信システムのそれぞれの通信装置のそれぞれのステータスまたはそれぞれのプロパティを記述することができる。この場合、各データレコードは、それぞれのエンティティのうちの一つと関連付けられてもよい。したがって、データレコードは全体として、1つの時点または1つの時間間隔におけるそれぞれのステータスを有する通信装置を記述してもよい。一例では、それぞれのエンティティのプロパティは、それぞれのエンティティに関連する年齢、性別、身長、収入、体重、出生地、もしくは職業、またはその組み合わせを含んでもよい。

10

【0019】

一例では、それぞれのエンティティ、例えばそれぞれの通信装置は、それぞれのデータレコードを生成することによって、訓練データを提供してもよい。別の例では、通信システムのロギングユニットは、それぞれのエンティティ、例えばそれぞれの通信装置の第1および第2の特徴量、もしくはターゲット特徴量、またはその両方の値を受信してもよく、これらそれぞれの値をそれぞれのデータレコードに書き込むことによって、それぞれのデータレコードを生成してもよい。

20

【0020】

ML-モジュールの訓練を実行した後、ML-モジュールは、訓練済み状態であってもよい。訓練済みML-モジュールは、少なくとも更なる特徴量の将来の値に依存するターゲット特徴量の将来の値を計算するために適用されてもよい。ターゲット特徴量の将来の値は、一般に、上述の機械の制御や、通信装置の1つまたは複数の動作の予測などのアプリケーションに用いられる値であってよい。したがって、以下では、ターゲット特徴量の将来の値を、ターゲット特徴量のアプリケーション値と呼ぶ。同様に、以下では、更なる特徴量の将来の値を、更なる特徴量のアプリケーション値と呼ぶ。

30

【0021】

一例では、訓練済みML-モジュールは、更なる特徴量のアプリケーション値をターゲット特徴量のアプリケーション値にマッピングするために設計された関数モジュールと見なされてもよい。ほとんどの場合、訓練済みML-モジュールは、ML-モジュールのいくつかのアプリケーション入力値をML-モジュールの出力のアプリケーション値にマッピングするために設計されてもよい。ML-モジュールの出力のアプリケーション値は、ターゲット特徴量のアプリケーション値を含んでもよい。アプリケーション入力値は、更なる特徴量のアプリケーション値を含んでもよい。更なる特徴量のアプリケーション値は、少なくとも第1もしくは第2またはその両方の特徴量のアプリケーション値に依存して決定されてもよい。アプリケーション入力値は、第1もしくは第2またはその両方の特徴量のアプリケーション値を含んでもよい。この場合、第1もしくは第2またはその両方の特徴量は、それぞれ訓練済みML-モジュールの第1および第2入力特徴量であってもよい。一般に、更なる特徴量は、ML-モジュールの更なる入力特徴量として考慮されてもよい。

40

【0022】

訓練データがセンサデータである例に従って参照すると、第1の特徴量のアプリケーション値および第2の特徴量のアプリケーション値は、それぞれ、第1のセンサによってキ

50

ャブチャされたアプリケーション第1センサ値および第2のセンサによってキャブチャされたアプリケーション第2センサ値であってよい。ターゲット特徴量のアプリケーション値は、アプリケーション第1センサ値、アプリケーション第2センサ値、もしくは更なる特徴量のアプリケーション値、またはその組み合わせに依存する機械を制御するための制御量を決定するために使用されてもよい。このように、訓練済みML-モジュールは、機械を制御するために機能してもよい。

【0023】

訓練データが1つまたは複数のエンティティ、例えば通信装置によって提供され得る例に従って参照すると、ターゲット特徴量のアプリケーション値は、エンティティのうちの1つの例示的なエンティティのアラーム信号が設定されているか、または設定されていないかのステータスであってもよい。第1の特徴量は、例示的なエンティティの第1のプロパティ、例えば、更なる通信装置へのアクティブな接続の数を記述してもよい。第2の特徴量は、例えば、所定の期間内に例示的なエンティティから送信されたメッセージの数など、例示的なエンティティの第2のプロパティを記述してもよい。この単純な適用例によれば、訓練済みML-モジュールは、少なくとも第1の特徴量のアプリケーション値、第2の特徴量のアプリケーション値、もしくは更なる特徴量のアプリケーション値、またはその組み合わせに依存する例示的なエンティティのアラーム信号のステータスを予測するために使用されてもよい。

10

【0024】

コンピュータシステムは、それぞれのデータレコードの第1のデータセットの値、もしくは第2のデータセットの値、またはその両方に数学的関数を適用することによって、各データレコードの更なるデータセットの各値を自動的に生成してもよい。例えば、コンピュータシステムは、各データレコードの更なるデータセットの各値を、それぞれのデータレコードの第1のデータセットの値および第2のデータセットの値の加重和、加重積または加重比として計算してもよい。第1の特徴量が年齢を記述し、第2の特徴量が収入を指す場合、コンピュータシステムは、各データレコードの更なるデータセットの各値を、各データレコードの第1のデータセットの値と第2のデータセットの値の比、例えば加重比として計算してもよい。この場合、第1のデータセットの値は、「1000」で重み付けされてもよい。

20

【0025】

更なるデータセットのそれぞれの値、すなわち更なる特徴量のそれぞれの値の、それぞれのデータレコードへの追加は、それぞれのデータレコードの次元を拡張することによって実行されてもよい。各データレコードの次元の数は、ML-モジュールの入力およびML-モジュールの出力の数と一致することがあるので、ML-モジュールの更なる入力は、それぞれのデータレコードに更なる特徴量のそれぞれの値を追加することに対応して提供されてもよい。更なる実施例によれば、更なる特徴量のそれぞれの値をそれぞれのデータレコードに追加することは、それぞれのデータレコードにおいて、選択された特徴量のそれぞれの値または後述する非選択特徴量のそれぞれの値を、更なる特徴量のそれぞれの値で置き換えることによって実行されてもよい。

30

【0026】

第1の相関メトリックは、ピアソン相関係数であってもよい。コンピュータシステムは、更なるデータセットと選択されたデータセットとに依存する第1の相関メトリックを計算してもよい。第1の閾値は、「0.5」から「0.9」の範囲内であってよく、例えば、「0.7」に等しいものであってよい。一例において、ML-モジュールは、第1の相関メトリックが「0.9」より大きい場合、更なるデータセットに基づいて訓練されないことがある。

40

【0027】

更なるデータセットを自動的に生成することは、関数セットから数学関数の自動選択を実行することを含んでもよい。一例において、コンピュータシステムは、選択をランダムに実行してもよい。一例において、コンピュータシステムは、更なるデータセットの生成

50

と、第1の相関メトリックの計算と、第1の相関メトリックが第1の閾値より大きいかどうかの確認と、更なるデータセットのそれぞれの値のそれぞれのデータレコードへの追加と、ML-モジュールの訓練の実行と、更なる特徴量の値の第1および第2のサブセットの定義と、バイアスメトリックの決定と、バイアスメトリックがバイアス制約を満たすかどうかの確認と、の繰り返しを実行してもよい。一例において、コンピュータシステムは、バイアスメトリックがバイアス制約を満たすまで、これらの繰り返しを実行してもよい。コンピュータシステムは、バイアスメトリックがバイアス制約を満たす場合、ML-モジュールを使用のために解放してもよい。繰り返しのうち単一の繰り返しの実行時に、コンピュータシステムは、数学関数、第1および第2のデータセットの値に対する更なるデータセットの値の依存性、第1の相関メトリックの計算方法、第1の閾値、ML-モジュールの構造パラメータの値、ML-モジュールの訓練を指定するパラメータの値、もしくは繰り返しのうち以前の繰り返しに関する更なる特徴量の値の第1および第2のサブセットの定義方法、またはその組み合わせを変更してもよい。第1および第2のデータセットの値への更なるデータセットの値の依存性を変化させることは、繰り返しのうち1回の繰り返しにおいて第1のデータセットの値に依存して更なるデータセットの値を生成することと、繰り返しのうち別の繰り返しにおいて第2のデータセットの値に依存する更なるデータセットの値を生成することと、繰り返しのうちさらに別の繰り返しにおいて第1のデータセットおよび第2のデータセットの値に依存する更なるデータセットの値を生成することと、を含んでもよい。コンピュータシステムは、ユーザによる干渉を受けることなく、完全に自動的に繰り返しを実行することができる。

10

20

【0028】

本明細書で使用する「モジュール」という用語は、「モジュール」に関連する機能を実行するため、または「モジュール」に関連する機能を実行した結果としての、既知または将来開発されるハードウェア、実行可能プログラムなどのソフトウェア、人工知能、ファジィ論理またはこれらの組合せを指す。

【0029】

アプリケーション入力値をML-モジュールの入力に送信することは、ターゲット特徴量のアプリケーション値を予測するためにML-モジュールを使用するための要求と考えることができる。アプリケーション入力値は、ML-モジュールのアプリケーション入力データセットと見なすことができる。ML-モジュールは、ニューラルネット、畳み込みニューラルネット、もしくは、ラジアル基底関数ネット、またはその組み合わせを含んでもよい。ML-モジュールは、ML-モジュールのモデルパラメータの値に依存するターゲット特徴量のアプリケーション値を含むML-モジュールの出力のアプリケーション値の計算を実行してもよい。ML-モジュールのモデルパラメータは、ML-モジュールの構造が定義され、訓練が完了すると、ニューラルネット、畳み込みニューラルネット、もしくは、ラジアル基底関数ネット、またはその組み合わせを指定することができる。モデルパラメータは、一例として、神経回路網の重みであってもよい。ML-モジュールの構造は、例えば、ニューロンの数、隠れ層の数などの構造パラメータの値によって指定されてもよい。モデルパラメータは、ML-モジュールの訓練中に変化してもよく、一方、構造パラメータは、ML-モジュールの訓練中に一定であってもよい。

30

40

【0030】

アプリケーション入力値およびML-モジュールの出力のアプリケーション値は、それぞれ実数値を含んでもよい。一例では、ML-モジュールの出力のアプリケーション値は、例えば、例示的なエンティティのアラーム信号が設定されているか否かを表現するために、バイナリフォーマットを有していてもよい。一例では、ターゲット特徴量のアプリケーション値は、アプリケーション入力データセットまたはアプリケーション入力データセットに対応するインスタントもしくはエンティティが2つまたはいくつかのクラスのいずれかに分類され得るかを確率で表してもよい。

【0031】

前述の例について、ML-モジュールの出力のアプリケーション値は、第1もしくは第

50

2 またはその両方の特徴量のアプリケーション値が与えられた場合に、例示的なエンティティのアラーム信号が設定されている、または設定されるべきであることがどの程度の確率であるかの信頼値であってよい。この例は、以下において、第1の使用例と呼ばれる。

【0032】

ML - モジュールの訓練を実行する前に、ML - モジュールは未訓練の状態であってもよい。未訓練の状態では、モデルパラメータの値は、ランダムな値に等しくてもよい。これは、ML - モジュールの初期化を実行することによって達成されてもよく、モデルパラメータの値は、ランダムな値に設定されてもよい。コンピュータシステムは、訓練データ、すなわち、データレコードに基づいて、ML - モジュールの訓練を実行してもよい。ML - モジュールは、訓練後、訓練済み状態となってもよい。

10

【0033】

第1および第2のデータセットの値のフォーマットは、第1および第2の特徴量のアプリケーション値のフォーマットと等しくてもよい。データレコードのターゲット特徴量の値とターゲット特徴量のアプリケーション値についても同様であってもよい。

【0034】

データレコードは、訓練済みML - モジュールが使用され得る分類問題についての情報を表すことができる。一例では、各データレコードのターゲット特徴量のそれぞれの値は、それぞれのデータレコードの更なるデータセットのそれぞれの値、もしくは第1のデータセットもしくは第2のデータセットまたはその両方、またはその両方を含む訓練入力データセット、およびそれによってそれぞれのデータレコードに対応する対応エンティティまたはインスタントがそれぞれのクラスに正しくグループ化することがどの程度の確率であるかの尺度であってよい。第1の使用例に関して、各データレコードのターゲット特徴量の値は、設定されているそれぞれのエンティティまたはそれぞれのインスタントのアラーム信号に対する信頼値をそれぞれ示すことができる。

20

【0035】

各データレコードの第1および第2の特徴量の値は、それぞれ、それぞれのデータレコードに対応するエンティティまたはインスタントが指定されてもよい。例えば、各データレコードの収入の値および年齢の値は、それぞれのデータレコードに対応するそれぞれのエンティティを指定してもよい。一例では、それぞれのエンティティは、それぞれの人物に関連してもよい。

30

【0036】

コンピュータシステムは、ML - モジュールのモデルパラメータの値が、ML - モジュールの訓練誤差を低減するように適応され得るように、ML - モジュールの訓練を実行してもよい。訓練誤差は、ML - モジュールによって計算されたML - モジュールの出力の計算された訓練出力値と、各データレコードのターゲット特徴量の値との偏差に基づいて計算されてもよい。ML - モジュールの出力の各訓練出力値は、少なくともそれぞれのデータレコードの更なる特徴量の値に基づいて計算されてもよく、したがって、それぞれのデータレコードに関連づけられてもよい。ML - モジュールの出力の訓練出力値は、ターゲット特徴量の値と同じ形式を有してもよい。ほとんどの場合、コンピュータシステムは、更なる特徴量の値と、それぞれのデータレコードの第1もしくは第2またはその両方のデータセットの値と、に依存するML - モジュールの出力のそれぞれの訓練出力値を計算してもよい。

40

【0037】

この偏差に基づいて、ML - モジュールのモデルパラメータの値を適応させることで、訓練誤差を低減することができる。訓練誤差が所定の閾値に達した場合、ML - モジュールは、訓練済みとみなされてもよく、訓練済み状態であってもよい。ML - モジュールのモデルパラメータの値の適応は、線形回帰、バックプロパゲーション、K - meansなどの、しばしば「機械学習」と呼ばれる1つまたは複数の訓練アルゴリズムを用いて実行されてもよい。それゆえ、「機械学習モジュール」と呼ばれる。ML - モジュールは、機械学習によって、および、その訓練済み状態によって訓練される、または、ML - モジ

50

ールの状態を、未訓練状態から中間訓練状態へ、または、訓練済み状態へ、または、中間訓練状態から訓練済み状態へ変更するために、機械学習は、ML-モジュールに適用されるかもしれない。ML-モジュールは、未訓練状態と訓練済み状態との間に、異なる訓練状態、例えば、中間訓練状態を有していてもよい。それぞれの異なる訓練状態において、ML-モジュールは、ML-モジュールのモデルパラメータの値の異なる組み合わせを含んでもよい。

【0038】

コンピュータシステムは、テストデータセットによってバイアスメトリックを決定してもよい。各テストデータセットは、アプリケーション入力データセットと同じ形式を有していてもよい。さらに、各テストデータセットは、更なる特徴量に対応する値と、アプリケーション入力データセットが第1の特徴量の値を含む場合に第1の特徴量に対応する値と、アプリケーション入力データセットが第2の特徴量の値を含む場合に第2の特徴量に対応する値とを含んでいてよい。さらに、テストデータセットおよびアプリケーション入力データセットは、それぞれ選択された特徴量の値を含んでいると仮定するが、これは必ずしもそうである必要はない。

10

【0039】

テストデータセットの生成は、テスト値の下限および上限を含む選択された特徴量のテスト値のセットの作成を含むことができる。選択された特徴量のテスト値は、第1または第2の特徴量のいずれが選択された特徴量であるかに応じて、データレコードの第1の特徴量の値に等しくてもよいし、データレコードの第2の特徴量の値に等しくてもよい。一例では、選択された特徴量のテスト値は、データレコードの値から独立して、例えば、ランダムに生成されてもよい。以下では、第1の特徴量と第2の特徴量のうち、選択された特徴量でない特徴量を非選択特徴量と呼ぶことがある。

20

【0040】

コンピュータシステムは、選択された特徴量のテスト値に依存し、かつ上述の数学関数に依存するテストデータセットの更なる特徴量の値を計算してもよい。

【0041】

一例において、コンピュータシステムは、選択された特徴量のテスト値および非選択特徴量のランダムに計算された値に依存し、かつ、上述の数学関数に依存するテストデータセットの更なる特徴量の値を計算してもよい。非選択特徴量のランダムに計算された値は、非選択特徴量の値の下限と非選択特徴量の値の上限との間に等しく分布していてもよい。テストデータセットの非選択特徴量の値は、一例において、非選択特徴量のランダムに計算された値と等しくてもよい。

30

【0042】

更なる例によれば、テストデータセットの非選択特徴量の値は、データレコードの非選択特徴量の値であってもよい。この場合、コンピュータシステムは、選択された特徴量のテスト値およびデータレコードの非選択特徴量の値に依存し、かつ、上述の数学関数に依存するテストデータセットの更なる特徴量の値を計算してもよい。テストデータセットは、例えば、非選択特徴量のランダムな値を使用することによって、任意の方法でML-モジュールの入力空間をカバーすることを可能にするかもしれない。

40

【0043】

一例では、更なる特徴量の値の第1および第2のサブセットは、更なる特徴量のサブセット閾値によって定義されてもよい。更なる特徴量のサブセット閾値より小さい更なる特徴量の各値は、更なる特徴量の値の第1のサブセットに関連付けられてもよい。同様に、更なる特徴量のサブセット閾値より大きいまたは更なる特徴量のサブセット閾値と等しい更なる特徴量の各値は、更なる特徴量の値の第2のサブセットに関連付けられてもよい。

【0044】

一例では、更なる特徴量のサブセット閾値は、ユーザ、例えば、ML-モジュールの生成を監督する監督者によって定義されてもよい。この場合、更なる特徴量のサブセット閾値は、選択された特徴量のサブセット閾値とは独立して定義されてもよい。

50

【 0 0 4 5 】

別の例によれば、コンピュータシステムは、選択された特徴量のサブセット閾値に依存し、かつ、上述の数学関数に依存する更なる特徴量のサブセット閾値を計算してもよい。したがって、更なる特徴量のサブセット閾値は、選択された特徴量のサブセット閾値の関数であってもよく、ここで、関数は、少なくとも選択された特徴量の値に依存する更なる特徴量の値を計算するためにも使用される。さらに、コンピュータシステムは、更なる特徴量の値が非選択特徴量の値に依存する場合、非選択特徴量の選択された値に依存する更なる特徴量のサブセット閾値を計算してもよい。選択された特徴量のサブセット閾値は、ユーザによって定義されてもよい。

【 0 0 4 6 】

コンピュータシステムは、テストデータセットの更なる特徴量の値に依存するテストデータセットを分割してもよい。一例では、テストデータセットは、第1および第2のグループの少なくとも2つのグループに分割されてもよい。第1のグループのテストデータセットは、それぞれ、更なる特徴量の値の第1のサブセットに関連付けられた、更なる特徴量の値を含んでもよい。同様に、第2のグループのテストデータセットは、それぞれ、更なる特徴量の値の第2のサブセットに関連付けられた更なる特徴量の値を含んでもよい。

【 0 0 4 7 】

バイアスメトリックは、バイアス最適化基準がどの程度まで満たされているかを確認するために使用されてもよい。バイアス最適化基準は、ユーザによって与えられてもよい。一例では、バイアス最適化基準は、ML-モジュールの出力の所定のテスト出力値を計算するための確率が、更なる特徴量の値から独立していることを含んでもよい。ML-モジュールの出力の所定のテスト出力値は、ML-モジュールの出力値がバイナリ値である場合、「1」に等しくてもよい。第1の使用例を参照すると、ML-モジュールの出力の所定のテスト出力値の値「1」は、例示的なエンティティのアラーム信号が設定されている、または設定されるべき状況に対応することができる。

【 0 0 4 8 】

バイアスメトリックを計算するために、コンピュータシステムは、訓練済みML-モジュールによって、および、それぞれのテストデータセットに依存する各テストデータセットに対するML-モジュールの出力の対応するテスト出力値を計算することができる。これを実現するために、それぞれのテストデータセットの値は、訓練済みML-モジュールに送信されてもよい。第1のグループのテストデータセットの1つに対応する訓練済みML-モジュールの出力の各テスト出力値に依存して、第1の確率が計算されてもよい。第1の確率は、アプリケーション入力データセットがテストデータセットの第1のグループに関連付けることができる場合、訓練済みML-モジュールの出力のアプリケーション出力値が「1」に等しい確率を示すことができる。第1の確率は、「1」に等しく、第1のグループのテストデータセットに対応するそれらのテスト出力値の数と、第1のグループのテストデータセットの総数と、の比に100%を乗じたものとして計算されてもよい。

【 0 0 4 9 】

同様に、コンピュータシステムは、第2のグループのテストデータセットのうちの1つに対応する各テスト出力値に依存する第2の確率を計算してもよい。第2の確率は、アプリケーション入力データセットがテストデータセットの第2のグループに関連付けることができる場合、訓練済みML-モジュールの出力のアプリケーション出力値が「1」に等しい確率を示してもよい。第2の確率は、「1」に等しく、第2のグループのテストデータセットに対応するそれらのテスト出力値の数と、第2のグループのテストデータセットの総数と、の比に100%を乗じたものとして計算されてもよい。

【 0 0 5 0 】

バイアスメトリックは、第1の確率と第2の確率との差に等しくてもよい。バイアスメトリックは、この場合、統計的なパリティ差に対応してもよい。別の例では、バイアスメトリックは、第1の確率と第2の確率の比に等しくてもよい。バイアスメトリックは、この場合、格差影響比率に対応してもよい。この場合、バイアス制約は、例えば、バイアス

10

20

30

40

50

メトリックが「0.9」よりも高いことであってもよい。一般に、バイアス制約は、バイアスメトリックが所定のバイアス閾値、例えば「0.9」よりも高い場合に、バイアス制約を満たすように設計されてもよい。

【0051】

更なる変形例では、コンピュータシステムは、第1のグループのテストデータセットによって計算された偽陽性テスト出力値の数（以下、第1の偽陽性数という）、および第2のグループのテストデータセットによって計算された偽陽性テスト出力値の数（以下、第2の偽陽性数という）に依存してバイアスメトリックを決定することができる。代替的に、または追加的に、コンピュータシステムは、第1のグループのテストデータセットによって計算された偽陰性テスト出力値の数（以下、第1の偽陰性数という）、および第2のグループのテストデータセットによって計算された偽陰性テスト出力値の数（以下、第2の偽陰性数という）に依存してバイアスメトリックを決定してもよい。バイアスメトリックは、第1の偽陽性数と第2の偽陽性数との間の比率に等しくてもよいし、第1の偽陰性数と第2の偽陰性数との間の比率に等しくてもよい。あるいは、バイアスメトリックは、第1の偽陽性数と第2の偽陽性数との間の比率と、第1の偽陰性数と第2の偽陰性数との間の比率との平均に等しくてもよい。なお、以下では、最後に述べたバイアスメトリックを分離基準と呼ぶことがある。

【0052】

一般に、少なくとも更なる特徴量の値に依存するML-モジュールの訓練を実行することは、例えば、訓練誤差を低減することによって、ML-モジュールの精度を向上させるのに役立つ場合がある。さらに、このことは、選択された特徴量に関するML-モジュールのバイアスを低減させることができる。データレコードに更なる特徴量の値を追加することによって、ML-モジュールの入力空間の次元が増加することがある。入力空間の次元のそのような増加は、選択された特徴量に関するML-モジュールのバイアスが、更なる特徴量の値を使用しないML-モジュールの訓練と比較して低減し得るように、ML-モジュールの訓練を軽減し得る。これは、以下のように説明することができる。更なる特徴量の値を使用することは、ML-モジュールの訓練のデータレコードの値の数を増加させる可能性がある。その結果、ML-モジュールのモデルパラメータの数は、訓練中のML-モジュールのオーバーフィッティングのリスクを伴わずに増加され得る。ML-モジュールのモデルパラメータの数が多いほど、一般的にバイアス最適化基準への到達が軽減され得るため、ML-モジュールのモデルパラメータの数が多いほど、訓練済みML-モジュールの低バイアスメトリックをもたらす機会を増加させることができる。

【0053】

この効果は、更なる特徴量の値が、選択されたデータセットの値と、第1のデータセットおよび第2のデータセットのうち選択されたデータセットではないデータセット（以下、非選択データセットという）の値とに依存して生成される場合、訓練中に増加することができるか、または少ない計算量で達成することができるかもしれない。これは、更なるデータセットの値を生成するために、非選択データセットと選択されたデータセットとを用いることにより、ML-モジュールの訓練中にML-モジュールのモデルパラメータの数をさらに増加させることができる場合があるためである。

【0054】

したがって、一実施形態によれば、更なるデータセットの生成は、選択されたデータセットおよび非選択データセットに依存する更なるデータセットを生成することを含む。

【0055】

バイアスメトリックを決定し、バイアスメトリックがバイアス制約を満たすかどうかを確認することは、更なる特徴量に関する訓練済みML-モジュールのバイアスが少しでも構築されているか、またはある許容範囲まで構築されているかをテストすることを可能にし得る。バイアスメトリックがバイアス制約を満たす場合、ML-モジュールは公正である、すなわち、ML-モジュールは許容範囲内で更なる特徴量の値の第1または第2のサブセットに偏っていないと仮定することができる。その結果、更なる特徴量の値の第1の

サブセットも第2のサブセットも、特権を与えられない場合がある。これにより、機械の異なる動作点、例えば第1のサブセットと第2のサブセットとによってカバーされる動作点における機械の制御において良好な性能を得ることができる場合がある。一方、これは、ML-モジュールが、選択された特徴量の値のサブセット、もしくは更なる特徴量の値の第1または第2のサブセット、またはその両方に対してバイアスを有することを防止することを可能にし得る。

【0056】

一例では、バイアスメトリックの決定およびバイアスメトリックがバイアス制約を満たすかどうかの確認は、第1の関連メトリックが第1の閾値より大きい場合にのみ実行されてもよい。これは、第1の関連メトリックの値が大きいほど、バイアスメトリックの値が高くなることを誘発し得るとする仮定によるものである。第1の関連メトリックの値が比較的低い場合、バイアスメトリックがバイアス制約を満たすかどうかを確認する必要がない場合がある。

10

【0057】

バイアスメトリックがバイアス制約を満たす場合、ML-モジュールは、例えば、上述のアプリケーション入力値に依存するターゲット特徴量のアプリケーション値を計算するために、使用のために解放されるかもしれない。

【0058】

上述の機械学習アルゴリズム、例えばバックプロパゲーションアルゴリズムは、ML-モジュールのそれぞれのモデルパラメータのそれぞれの値に関するバイアス最適化基準のそれぞれの導関数によって、ML-モジュールの各モデルパラメータの値のそれぞれの変化を計算するために使用されてもよい。一例では、バイアス最適化基準は、「1」と第1の確率と第2の確率の比との間の差のノルムに等しくてもよい。このノルムは、第1の確率と第2の確率の比が「1」に等しくなるように最小化されることが望ましい。一例では、このノルムを、上述した訓練誤差に加算してもよい。この場合、ML-モジュールの訓練の勾配ベースの各反復において、訓練誤差とバイアス最適化基準との和が減少してもよい。

20

【0059】

一実施形態によれば、更なるデータセットの生成は、選択されたデータセットに依存する更なるデータセットを生成することを含んでもよい。これにより、選択されたデータセットの値の関数として与えられるアプリアリ条件付き確率をより正確に近づけることができる場合がある。この効果は、バイアスメトリックが上述の分離基準と同様または等しい場合に有用であり得る。また、この効果は、更なるデータセットが、選択されたデータセットと非選択データセットとに依存して生成される場合にも得られる可能性がある。更なるデータセットの生成のために非選択データセットを使用することは、選択された特徴量もしくは更なる特徴量またはその両方に関するML-モジュールのバイアスを低減するために、ML-モジュールのより多くの数のモデルパラメータでML-モジュールを訓練することを追加的に可能にし得る。

30

【0060】

一実施形態によれば、更なるデータセットの生成は、非選択データセットに依存する更なるデータセットを生成することを含んでもよい。この実施形態によれば、更なるデータセットの生成は、選択されたデータセットとは独立して実行されてもよい。この場合、訓練済みML-モジュールが第1のサブセットまたは第2のサブセットのいずれかに偏る可能性があるが、バイアスメトリックがバイアス制約を満たすかどうかを確認することにより、ML-モジュールを使用するために解放することを防止することができる。これは、ML-モジュールの生成が完全に自動化された場合に特に有効である。

40

【0061】

一実施形態によれば、選択された特徴量は、ユーザによって選択されてもよい。この実施形態によれば、方法は、非選択データセットと選択されたデータセットとに依存する非選択特徴量と選択された特徴量との間の関連の尺度として、第2の関連メトリックを計算

50

することをさらに含んでもよい。方法は、第2の相関メトリックが第2の閾値よりも大きいかどうかを確認することをさらに含んでもよい。

【0062】

さらに、ML-モジュールの訓練は、少なくとも非選択データセットに依存するML-モジュールを訓練することを含んでもよい。例えば、ML-モジュールの訓練は、ML-モジュールによって、およびそれぞれのデータレコードの非選択データセットの値に依存して、各データレコードのML-モジュールの出力のそれぞれの訓練出力値を計算することを含む場合がある。方法は、少なくとも非選択特徴量の値の第1のサブセットと、非選択特徴量の値の第2のサブセットとを定義することをさらに含むことができる。非選択特徴量の値の第1のサブセットおよび第2のサブセットは、不連続なサブセットであってもよい。それに加えて、第2の相関メトリックが第2の閾値より大きい場合、非選択特徴量の値の第1のサブセットまたは非選択特徴量の値の第2のサブセットに対するML-モジュールのバイアスの強さを示す更なるバイアスメトリックが決定されてもよい。

10

【0063】

さらに、コンピュータシステムは、更なるバイアスメトリックが更なるバイアス制約を満たすかどうかを確認してもよい。更なるバイアスメトリックが更なるバイアス制約を満たす場合、ML-モジュールは使用のために解放されてもよい。この実施形態の利点は、訓練済みML-モデルが偏ることを誘発し得る、可能性のある全ての特徴量、ここでは第1および第2の特徴量間の選択に関するユーザの不完全な選択が、自動的に補償され得ることであり得る。第2の相関メトリックが第2の閾値よりも大きい場合、これは、訓練済みML-モジュールが、非選択特徴量の値の第1のサブセットまたは非選択特徴量の値の第2のサブセットのいずれかに対するバイアスを含むリスクを増加させ得る、すなわち、訓練済みML-モジュールが「不公平」である可能性がある。この場合、更なるバイアスメトリックが決定され、更なるバイアス閾値に対して確認されるので、本実施形態によれば、ユーザの不完全な選択が補償され得る。更なるバイアスメトリックの決定および更なるバイアスメトリックが更なるバイアス制約を満たすかどうかの確認は、バイアスメトリックの決定およびバイアスメトリックがバイアス制約を満たすかどうかの確認と同様に行われてもよいが、更なるデータセットを用いる代わりに非選択データセットが用いられ、バイアス閾値の代わりに更なるバイアス閾値が適用されてもよい。

20

【0064】

一実施形態によれば、方法は、第1の相関メトリックを計算すること、もしくは第1の相関メトリックがユーザ相関入力データに依存する第1の閾値よりも大きいかどうかを確認すること、またはその両方を行うことをさらに含んでもよい。ユーザ相関入力データは、第1の閾値を含んでもよい。したがって、本実施形態は、訓練済みML-モジュールがそのバイアスに関してテストされるかどうかによりユーザが影響力を持つことを可能にし得る。第1の閾値が高いほど、訓練済みML-モジュールがそのバイアスに関してテストされる確率は低くてもよい。

30

【0065】

一実施形態によれば、方法は、少なくとも更なる特徴量の値の第1のサブセットと、ユーザバイアス入力データに依存する更なる特徴量の値の第2のサブセットと、を定義することをさらに含んでもよい。例えば、ユーザバイアス入力データは、更なる特徴量のサブセット閾値もしくは更なる特徴量の値の第1のサブセットの下限値またはその両方、もしくは、更なる特徴量の値の第2のサブセットの上限値、またはその両方を含んでもよい。したがって、本実施形態は、ユーザがバイアスメトリックの計算に影響を与えることを可能にし得る。

40

【0066】

一実施形態によれば、方法は、それぞれ少なくとも更なる特徴量に対応する値を含むアプリケーション入力データセットに依存するML-モジュールの出力の更なる値を計算するために訓練済みML-モジュールを使用するとき、更なる特徴量に関する訓練済みML-モジュールのバイアスを監視するためのバイアスメトリックを訓練済みML-モジュール

50

ルのメタデータの第1の部分の形態で記憶することをさらに含む場合がある。アプリケーション入力データセットは、上述のアプリケーション入力データセットと同様に構成されてもよく、また設計されてもよい。方法は、更なる特徴量に関するML-モジュールのバイアスが増加する場合、ML-モジュールの使用を中止することを更にも含む。更なる特徴量に関するML-モジュールのバイアスの変化は、上述のようにバイアスメトリックを再び計算することと、データレコードの値の代わりにまたは追加的にアプリケーション入力データセットの値を使用することとによって検出されてもよい。

【0067】

一実施形態によれば、方法は、訓練済みML-モジュールのメタデータの第2の部分として第1の相関メトリックを記憶することをさらに含む。第1の相関メトリックは、それぞれ少なくとも更なる特徴量に対応する値を含むアプリケーション入力データセットに依存するML-モジュールの出力の更なる値を計算するために訓練済みML-モジュールを使用するとき、更なる特徴量に関する訓練済みML-モジュールのバイアスを監視するために使用され得る。

10

【0068】

一例では、選択された特徴量と更なる特徴量との間の相関の尺度としての更なる第1の相関メトリックは、アプリケーション入力データセットの選択された特徴量の値と、アプリケーション入力データセットに依存して計算された更なる特徴量の値とに依存して計算されてもよい。更なる第1の相関メトリックは、第1の相関メトリックと比較されてもよい。更なる第1の相関メトリックが第1の相関メトリックより低い場合、バイアスメトリックは再計算されないことがあり、ML-モジュールの出力の更なる値の計算のために訓練済みML-モジュールを使用するとき、更なる特徴量に関する訓練済みML-モジュールのバイアスは確認されないことがある。一方、更なる第1の相関メトリックが第1の相関メトリックよりも高い場合、バイアス制約が厳しくなってもよい。例えば、バイアス閾値が増加されてもよい。

20

【0069】

一実施形態によれば、方法は、それぞれ少なくとも非選択特徴量に対応する値を含むアプリケーション入力データセットに依存するML-モジュールの出力の更なる値を計算するために訓練済みML-モジュールを使用するとき、非選択特徴量に関する訓練済みML-モジュールのバイアスを監視するための更なるバイアスメトリックを訓練済みML-モジュールのメタデータの第3の部分として記憶することをさらに含む。方法は、非選択特徴量に関するML-モジュールのバイアスが増加する場合、ML-モジュールの使用を中止することをさらに含む。非選択特徴量に関するML-モジュールのバイアスの変化は、再び上述のように更なるバイアスメトリックを計算することと、データレコードの値の代わりにまたは追加的にアプリケーション入力データセットの値を使用することとによって検出され得る。

30

【0070】

一実施形態によれば、方法は、それぞれ少なくとも非選択特徴量に対応する値を含むアプリケーション入力データセットに依存するML-モジュールの出力の更なる値を計算するために訓練済みML-モジュールを使用するとき、非選択特徴量に関する訓練済みML-モジュールのバイアスを監視するための第2の相関メトリックを訓練済みML-モジュールのメタデータに第4の部分として記憶することをさらに含む。

40

【0071】

一例において、選択された特徴量と非選択特徴量との間の相関の尺度としての更なる第2の相関メトリックは、アプリケーション入力データセットの選択された特徴量の値とアプリケーション入力データセットの非選択特徴量の値とに依存して計算されてもよい。更なる第2の相関メトリックは、第2の相関メトリックと比較されてもよい。更なる第2の相関メトリックが第2の相関メトリックより低い場合、更なるバイアスメトリックは再計算されなくてもよく、ML-モジュールの出力の更なる値の計算のために訓練済みML-モジュールを使用するとき、非選択特徴量に関する訓練済みML-モジュールのバイアス

50

は確認されなくてもよい。一方、更なる第2の相関メトリックが第2の相関メトリックよりも高い場合、更なるバイアス制約が厳しくなってもよい。例えば、更なるバイアス閾値は、増加されてもよい。

【0072】

一実施形態によれば、方法は、ML-モジュールの構造を修正することと、ML-モジュールの訓練の実行を繰り返すことと、バイアスメトリックの計算を繰り返すことと、バイアスメトリックが減少したかどうかを確認することと、を更に含んでもよい。バイアスメトリックが減少していた場合、修正された構造を持つML-モジュールは使用のために解放されてもよく、以前のバージョンのML-モジュールは拒否されてもよい。

【0073】

ML-モジュールの構造は、構造パラメータで指定することができる。構造パラメータは、ML-モジュールが神経回路網を含む場合、隠れ層の数、ニューロンの数、またはニューロンの活性化関数のタイプを含んでもよく、ML-モジュールがサポートベクターマシンを含む場合、カーネルのタイプを含んでもよい。したがって、構造の修正は、ML-モジュールが神経回路網を含む場合、隠れ層の数、ニューロンの数、もしくは活性化関数のタイプ、またはその組み合わせを変更することを含んでいてもよく、ML-モジュールがサポートベクターマシンを含む場合、カーネルのタイプを変更することを含んでいてもよい。ML-モジュールの構造の変更は、選択された特徴量もしくは更なる特徴量またはその両方に関して、訓練されたML-モジュールのバイアスをさらに低減させることを可能にし得る。

【0074】

一実施形態によれば、方法は、ML-モジュールの構造を修正することと、ML-モジュールの訓練を実行することと、バイアスメトリックを計算することと、の複数の繰り返しを実行することをさらに含んでもよい。方法は、それぞれの反復について、ML-モジュールの構造を指定する構造パラメータの値のそれぞれのセットを記録することと、更なるML-モジュールのバイアスを低減させることに関して更なるML-モジュールの構造を最適化するためのデータベースを提供するために、構造パラメータの値のセットをそれぞれのバイアスメトリックと共に記憶することと、をさらに含むことができる。一例では、ML-モジュールの構造を指定する構造パラメータの値のセットのそれぞれは、構造の修正の各反復において作成されたML-モジュールの構造の隠れ層の数、ニューロンの数、もしくは活性化関数のタイプ、またはその組み合わせから構成されてもよい。データベースは、更なるML-モジュールの構造パラメータの適切な値を見つけることを軽減してもよい。

【0075】

本発明の実施形態は、コンピュータシステム、クライアント、またはサーバとも呼ばれることがあるコンピューティング装置を用いて実施することができる。ここで、図1を参照すると、コンピュータシステム1の一例の概略図が示されている。コンピュータシステム1は、好適なコンピュータシステムの一例に過ぎず、本明細書に記載する本発明の実施形態の使用や機能の範囲について何らの限定も示唆するものではない。いずれにせよ、コンピュータシステム1は、実装されること、もしくは上述した機能のいずれかを行うこと、またはその両方が可能である。

【0076】

コンピュータシステム1は、図1に示すように、第1のコンピュータシステム/サーバ12を含む。

【0077】

第1のコンピュータシステム/サーバ12は、他の数多くのコンピューティングシステム環境または構成と共に動作可能である。第1のコンピュータシステム/サーバ12と共に使用するのに適している可能性のある周知のコンピューティングシステム、環境もしくは構成またはその組み合わせの一例としては、パーソナルコンピュータシステム、サーバコンピュータシステム、シンクライアント、シッククライアント、ハンドヘルドまたは

10

20

30

40

50

ラップトップ装置、マルチプロセッサシステム、マイクロプロセッサベースのシステム、セットトップボックス、プログラマブル家電製品、ネットワークPC、ミニコンピュータシステム、メインフレームコンピュータシステム、および上記のシステムまたは装置のいずれかを含む分散型クラウドコンピューティング環境などが挙げられる。

【0078】

第1のコンピュータシステム/サーバ12は、コンピュータシステムによって実行されるプログラムモジュールなどの、コンピュータシステム実行可能命令との一般的な関連において説明することができる。一般に、プログラムモジュールは、特定のタスクを実行するかまたは特定の抽象データ型を実装するルーチン、プログラム、オブジェクト、コンポーネント、ロジック、データ構造などを含むことができる。第1のコンピュータシステム/サーバ12は、通信ネットワークを介してリンクされたりリモート処理装置によってタスクが実行される分散型クラウドコンピューティング環境で実施することができる。分散型クラウドコンピューティング環境において、プログラムモジュールは、メモリ記憶装置を含む、ローカルおよびリモート両方のコンピュータシステム記憶媒体に配置されてもよい。

10

【0079】

図1に示すように、コンピュータシステム1における第1のコンピュータシステム/サーバ12を、汎用コンピュータ装置として示している。コンピュータシステム/サーバ12のコンポーネントの一例としては、1つ以上のプロセッサまたは処理ユニット16、システムメモリ28、およびシステムメモリ28を含む種々のシステムコンポーネントをプロセッサ16に接続するバス18が挙げられる。バス18は、種々のバスアーキテクチャのいずれかを使用するメモリバスまたはメモリコントローラ、周辺バス、アクセラレーテッドグラフィックスポート(AGP)、およびプロセッサまたはローカルバスを含む複数種類のバス構造のうち1つ以上の任意のものを表す。一例として、かかるアーキテクチャは、インダストリスタンダードアーキテクチャ(ISA)バス、マイクロチャンネルアーキテクチャ(MCA)バス、拡張ISA(EISA)バス、ビデオエレクトロニクススタンダードアソシエーション(VESA)ローカルバス、およびペリフェラルコンポーネントインターコネクト(PCI)バスを含む。

20

【0080】

第1のコンピュータシステム/サーバ12は一般的に、種々のコンピュータシステム可読媒体を含む。かかる媒体は、コンピュータシステム/サーバ12によってアクセス可能な任意の利用可能な媒体でよく、揮発性媒体および不揮発性媒体の両方と、取り外し可能媒体および取り外し不能媒体の両方とを含む。

30

【0081】

システムメモリ28は、ランダムアクセスメモリ(RAM)31もしくはキャッシュメモリ32またはその両方など、揮発性メモリとしてのコンピュータシステム可読媒体を含むことができる。第1のコンピュータシステム/サーバ12はさらに、他の取り外し可能/取り外し不能コンピュータシステム可読媒体および揮発性/不揮発性コンピュータシステム可読媒体を含んでもよい。一例として、ストレージシステム34は、取り外し不能な不揮発性磁気媒体(不図示。一般に「ハードドライブ」と呼ばれる)への読み書きのために設けることができる。また、図示は省略するが、取り外し可能な不揮発性磁気ディスク(例えば、フロッピーディスク)への読み書きのための磁気ディスクドライブ、および取り外し可能な不揮発性光学ディスク(CD-ROM、DVD-ROMや他の光学媒体など)への読み書きのための光学ディスクドライブを設けることができる。これらの例において、それぞれを、1つ以上のデータ媒体インタフェースによってバス18に接続することができる。以下でさらに図示および説明するように、メモリ28は、本発明の実施形態の機能を実行するように構成されたプログラムモジュールのセット(例えば、少なくとも1つ)を有する少なくとも1つのプログラム製品を含むことができる。

40

【0082】

一例として、プログラムモジュール50のセット(少なくとも1つ)を有するプログラ

50

ム/ユーティリティ35は、オペレーティングシステム、1つ以上のアプリケーションプログラム、他のプログラムモジュール、およびプログラムデータと同様に、メモリ28に記憶することができる。オペレーティングシステム、1つ以上のアプリケーションプログラム、他のプログラムモジュール、およびプログラムデータ、またはそれらのいくつかの組み合わせの各々は、ネットワーク環境の実装形態を含むことができる。プログラムモジュール50は一般に、本発明の実施形態の機能もしくは方法またはその両方を実行する。

【0083】

また、第1のコンピュータシステム/サーバ12は、キーボード、ポインティングデバイス、ディスプレイ24などの1つ以上の外部装置14、ユーザと第1のコンピュータシステム/サーバ12との対話を可能にする1つ以上の装置、もしくは第1のコンピュータシステム/サーバ12と1つ以上の他のコンピュータ装置との通信を可能にする任意の装置（例えば、ネットワークカードやモデムなど）またはこれらの組み合わせと通信することができる。かかる通信は、入力/出力(I/O)インタフェース22を介して行うことができる。さらに、第1のコンピュータシステム/サーバ12は、ネットワークアダプタ25を介して1つ以上のネットワーク（ローカルエリアネットワーク(LAN)、汎用広域ネットワーク(WAN)、もしくはパブリックネットワーク（例えばインターネット）またはこれらの組み合わせなど）と通信することができる。図示するように、ネットワークアダプタ25は、バス18を介して第1のコンピュータシステム/サーバ12の他のコンポーネントと通信する。なお、図示は省略するが、他のハードウェアコンポーネントもしくはソフトウェアコンポーネントまたはその両方を、第1のコンピュータシステム/サーバ12と併用することができる。それらの一例としては、マイクロコード、デバイスドライバ、冗長化処理ユニット、外付けディスクドライブアレイ、RAIDシステム、テープドライブ、データアーカイブストレージシステムなどがある。

【0084】

図1に示すコンピュータシステム1などのコンピュータシステムは、以下において第1の機能とも呼ばれる訓練データ100の提供などの本明細書に開示される動作を実行するように構成されてもよい。一例では、コンピュータシステム1は、プロセッサ16上で訓練データ検索モジュール51を実行することによって、訓練データ100を提供してもよい。データ検索モジュール51の実行は、ストレージシステム34からRAM31への訓練データ100のロードを誘発してもよい。訓練データ100は、第1のデータセット10と、第2のデータセット20と、ターゲットデータセット30とを含んでもよい。第1のデータセット10は、第1の特徴量に対応する値 10_1 、 10_2 、 10_i 、 10_n を含んでもよい。第2のデータセット20は、第2の特徴量に対応する値 20_1 、 20_2 、 20_i 、 20_n を含んでもよい。ターゲットデータのセット30は、ターゲット特徴量に対応する値 30_1 、 30_2 、 30_i 、 30_n を含んでもよい。したがって、第1のデータセット10、第2のデータセット20、およびターゲットデータのセット30は、それぞれ、第1の特徴量、第2の特徴量、およびターゲット特徴量に対応してもよい。一例において、第1の特徴量および第2の特徴量の値は、上述したように、通信システム300のそれぞれのエンティティ 300_i のプロパティをそれぞれ記述していてもよい。ターゲット特徴量の値は、上述した第1の使用例に従って、それぞれのエンティティ 300_i のステータス、またはそれぞれのエンティティ 300_i によって実行されているアクションまたはエンティティ 300_i とともに実行されているアクション、例えばそれぞれのエンティティ 300_i のアラーム信号またはそれぞれのエンティティ 300_i への参照が設定されているか否か、をそれぞれ記述していてもよい。

【0085】

訓練データ100は、それぞれのデータレコード110（ 110_1 、 110_2 、 110_i 、 110_n ）に構造化されてもよい。データレコードは、ストレージシステム34のデータベース120に記憶されてもよい。各データレコード 110_i は、第1のデータセット10のそれぞれの値 10_i 、すなわち、第1の特徴量のそれぞれの値 10_i を含んでもよい。さらに、各データレコード 110_i は、第2のデータセット20のそれぞれの値2

0_iすなわち、第2の特徴量のそれぞれの値20_iを含んでもよい。一例では、各データレコード110_iは、ターゲットデータセット30のそれぞれの値30_i、すなわちターゲット特徴量のそれぞれの値30_iを含んでもよい。第1特徴量と第2特徴量の一方は、選択された特徴量であってもよい。以下では、第1の特徴量が選択された特徴量であると仮定する。したがって、第1のデータセット10は、選択されたデータセットとみなされてもよい。さらに、以下では、第2の特徴量が非選択特徴量であると仮定してもよい。同様に、第2のデータセット20は、非選択データセットとして考慮されてもよい。しかしながら、更なる実施例では、第2の特徴量は、更なる選択された特徴量であってもよい。第1の特徴量は、例えば、1つまたは複数の外部装置14、例えばキーボードによって、第1の特徴量の選択を参照する入力データを供給することによって、コンピュータシステム1のユーザによって選択されてもよい。

【0086】

コンピュータシステム1は、以下において第2の機能とも呼ばれる、第1のデータセット10および第2のデータセット20の少なくとも1つのデータセットに依存する更なるデータセット40を自動的に生成してもよい。更なるデータセット40は、更なる特徴量に対応してもよい。更なるデータセット40は、更なる特徴量の値40₁、40₂、40_i、40_nを含んでもよい。プロセッサ16は、更なるデータセット40のそれぞれの値40_iを、第1のデータセット10のそれぞれの値10_iもしくは第2のデータセット20のそれぞれの値20_iまたはその両方に依存して生成するために、関数のセットから数学関数を選択してもよく、使用してもよい。プロセッサ16は、第2の関数を実行するための関数モジュール52を実行してもよい。

【0087】

単純な例では、第1の特徴量は、年齢であってもよい。したがって、それぞれの値10_iは、それぞれのエンティティ300_iに対応するそれぞれの年齢を記述してもよい。さらに、この単純な例によれば、第2の特徴量は、収入であってもよい。したがって、それぞれの値20_iは、それぞれのエンティティ300_iに対応するそれぞれの収入を記述してもよい。一例において、更なる特徴量のそれぞれの値40_iを決定するための数学的関数は、以下の通りであってもよい。

$$40_i = (10_i * 1000) / 20_i$$

30

【0088】

更なる例では、更なる特徴量のそれぞれの値40_iは、選択された特徴量の値10_iにのみ依存して計算されてもよい。この場合、更なる特徴量のそれぞれの値40_iを決定するための数学的関数は、以下の通りであってもよい。

$$40_i = \lceil \ln(4 + 10_i) \rceil$$

【0089】

数学的関数としての対数関数は、選択された特徴量のそれぞれの値10_iが、約0.25未満の値を有する選択された特徴量のそれぞれの値10_iに関する導関数を有していてもよいことを誘発し得る。

【0090】

コンピュータシステム1は、図3に示すように、更なる特徴量のそれぞれの値40_iをそれぞれのデータレコード110_iに追加するように構成されていてもよい。例えば、RAM31は、更なる特徴量のそれぞれの値40_iをデータベース120のそれぞれのデータレコード110_iに書き込むための書き込みコマンドを含むメインプログラムを記憶していてもよい。

【0091】

コンピュータシステム1は、更なるデータセット40および選択されたデータセットに

50

依存する選択された特徴量と更なる特徴量との間の相関の尺度として、第1の相関メトリックを計算してもよい。プログラムモジュール50は、選択された特徴量および更なる特徴量の共分散と、選択されたデータセットおよび更なるデータセット40に依存する選択された特徴量の標準偏差および更なる特徴量の標準偏差の積との間の比率としてピアソン相関係数を計算するための統計モジュール53を含んでいてもよい。

【0092】

プロセッサ16は、第1の相関メトリックが第1の閾値、例えば「0.7」よりも大きいかどうかを確認するように構成されてもよい。これを実現するために、メインプログラムは、比較演算子を含んでいてもよい。

【0093】

コンピュータシステム1は、データレコード110を用いて、図4に示すML-モジュール400の訓練を行ってもよい。訓練は、訓練データ100に基づいて、ML-モジュール400の出力410の訓練出力値440(440₁、440₂、440_i、440_n)を計算することを含んでもよい。一般に、訓練は、ML-モジュール400によって、および少なくともそれぞれのデータレコード110_iの更なるデータセット40の値40_iに依存して、各データレコード110_iのそれぞれの訓練出力値440_iを計算することを含んでもよい。

【0094】

図4に示す例によれば、コンピュータシステム1は、訓練を実行するためのML-モジュール400の入力に、訓練入力データセット450の形態で第1のデータセット10、第2のデータセット20、および更なるデータセット40の値を送信してもよい。プロセッサ16は、第1の特徴量のそれぞれの値10_i、すなわち第1のデータセット10のそれぞれの値10_iに依存し、第2の特徴量のそれぞれの値20_i、すなわち第2のデータセット20のそれぞれの値20_iに依存し、更なる特徴量のそれぞれの値40_i、すなわち更なるデータ40のそれぞれの値40_iに依存し、かつML-モジュール400のモデルパラメータ430の値に依存する、それぞれの訓練出力値440_iを計算してもよい。一例において、モデルパラメータ430は、ML-モジュール400が神経回路網である場合、ML-モジュール400の入力ニューロン401、402、403の1つとML-モジュール400の隠れラガーのニューロン420の1ニューロンとの間のそれぞれの接続の強さを示す重みであり得る。この場合、入力ニューロン401、402、403は、ML-モジュール400の入力を構築することができる。

【0095】

プロセッサ16は、それぞれの訓練出力値440_iを計算するために、第1の特徴量のそれぞれの値10_i、第2の特徴量のそれぞれの値20_iおよび更なる特徴量のそれぞれの値40_iをそれぞれの訓練入力データセット450_iの形態でML-モジュール400の入力に送信してもよい。

【0096】

コンピュータシステム1は、ML-モジュール400の訓練を実行するための訓練モジュール54を含んでいてもよい。訓練モジュール54の実行は、データレコード110をロードすることと、データレコード110に基づいて訓練入力データセット450を生成することと、訓練入力データセット450をML-モジュール400の入力に送ることと、ML-モジュール400のいくつかの訓練反復を実行することと、を含んでもよい。

【0097】

単一の訓練反復は、訓練出力値440を計算することと、訓練出力値440およびターゲット特徴量の値30に依存する訓練誤差を計算することと、訓練誤差に依存するモデルパラメータ430の各モデルパラメータの値のそれぞれの変化を計算することと、モデルパラメータ430の各モデルパラメータの値のそれぞれの変化に従ってモデルパラメータ430の各モデルパラメータの各値を適応させることと、を含んでいてもよい。訓練誤差は、四分位の合計であってよく、各四分位は、それぞれの訓練出力値440_iとそれぞれのデータレコード110_iのターゲット特徴量のそれぞれの値30_iとの間のそれぞれの

10

20

30

40

50

差の四分位である。

【0098】

プロセッサ16は、モデルパラメータ430のそれぞれのモデルパラメータの値のそれぞれの変化を、モデルパラメータ430のそれぞれのモデルパラメータに関する訓練誤差のそれぞれの導関数を計算することによって計算してもよい。プロセッサ16は、訓練誤差が訓練誤差閾値を下回るまで、単一の訓練反復のいくつかの実行を繰り返してもよい。この訓練基準が満たされる場合、ML-モジュール400は、訓練済み状態であってよい。

【0099】

さらに、プロセッサ16は、更なる特徴量の値の第1のサブセット501と、更なる特徴量の値の第2のサブセット502とを定義してもよく、第1のサブセット501と第2のサブセット502は不連続なサブセットである。一例では、プロセッサ16は、更なる特徴量510のサブセット閾値とデータレコード110の更なる特徴量の値 40_i とによって、第1のサブセット501と第2のサブセット502を定義してもよい。更なる特徴量510の閾値より小さい更なるデータセット40のすべての値は、第1のサブセット501の一部であってよく、更なる特徴量510の閾値より大きいまたは閾値と等しい更なるデータセット40のすべての値は、第2のサブセット502の一部であってよい。プロセッサ16は、第1のサブセット501および第2のサブセット502を生成するために、図5において矢印520で示される昇順で更なるデータセット40の値を並べ替えてもよい。例えば、更なるデータセット40の値 40_k は、第2のサブセット502の最も低い値であってよく、更なるデータセット40の値 40_{k-1} は、第1のサブセット501の最も高い値であってよい。

【0100】

図に示されていない別の例では、プロセッサ16は、更なる特徴量510のサブセット閾値および更なる特徴量の任意生成値によって、第1のサブセット501および第2のサブセット502を定義してもよい。この例によれば、更なる特徴量510の閾値より小さい更なる特徴量の任意生成値の全ての値は、第1のサブセット501の一部であってよく、更なる特徴量510の閾値より大きいまたは閾値と等しい更なる特徴量の任意生成値の全ての値は、第2のサブセット502の一部であってよい。

【0101】

さらに、プロセッサ16は、第1の相関メトリックが第1の閾値よりも大きい場合、更なる特徴量の値の第1のサブセット501または更なる特徴量の値の第2のサブセット502に対するML-モジュール400のバイアスの強さを示すバイアスメトリックを決定してもよい。

【0102】

コンピュータシステム1は、テストデータセット200 (200_1 、 200_2 、 200_i 、 200_m)を提供してもよい。各テストデータセット 200_i は、第1の特徴量のそれぞれの値 210_i と、第2の特徴量のそれぞれの値 220_i と、更なる特徴量のそれぞれの値 240_i とを含んでいてもよい。明確にするために、テストデータセット210の第1の特徴量の第1の値 210_1 、第2の値 210_2 およびm番目の値 210_m 、テストデータセット210の第2の特徴量の第1の値 220_1 、第2の値 220_2 およびm番目の値 220_m 、およびテストデータセット210の更なる特徴量の第1の値 240_1 、第2の値 240_2 およびm番目の値 240_m は図6に描かれている。一例では、コンピュータシステム1は、上述したように、データレコード110から独立してテストデータセット200の値を生成してもよい。プロセッサ16は、テストデータセット200の第1および第2の特徴量の値のランダムセットを生成してもよく、テストデータセット200の第1および第2の特徴量の値のランダムセットと数学関数とに依存するテストデータセット200の更なる特徴量の値を計算してもよい。

【0103】

簡単にするために、以下では、テストデータセット200の値が訓練入力データセット

450の値と等しいという変形例を説明する。したがって、この例によれば、テストデータセット200は、訓練入力データセット450と等しくてもよい。

【0104】

コンピュータシステム1は、第1のサブセット501と第2のサブセット502との2つのグループへの更なる特徴量の値の分割に従って、テストデータセット200を分割してもよい。言い換えれば、コンピュータシステム1は、更なる特徴量の値の各サブセットに、テストデータセット200の1つの対応するグループが提供され得るように、テストデータセット200を分割してもよい。一例では、テストデータセット200の第1のグループのテストデータセットはそれぞれ、第1のサブセット501に関連付けられる更なる特徴量の値を含んでいてもよい。同様に、テストデータセット200の第2のグループのテストデータセットは、それぞれ、第2のサブセット502に関連付けられた更なる特徴量の値を含んでいてもよい。

10

【0105】

バイアスメトリックは、バイアス最適化基準が満たされているか、またはどの程度満たされているかを確認するために使用されてもよい。バイアス最適化基準は、ユーザによって与えられてもよい。一例では、バイアス最適化基準は、訓練済みML-モジュール400の出力410の所定のテスト出力値を計算するための確率が、更なる特徴量の値から独立していることを含んでもよい。訓練済みML-モジュール400の出力410の所定のテスト出力値は、訓練済みML-モジュールの出力値がバイナリ値である場合、「1」に等しくてもよい。第1の使用例を参照すると、訓練済みML-モジュール400の出力410の所定のテスト出力値の値「1」は、アラーム信号が設定されている、または設定されるべき状況に対応し得る。

20

【0106】

バイアスメトリックを計算するために、コンピュータシステム1は、訓練済みML-モジュール400によって、それぞれのテストデータセット200_iに対して、訓練済みML-モジュール400の出力410の、および、それぞれのテストデータセット200_iに依存する、対応するテスト出力値700_iを計算してもよい。これを実現するために、コンピュータシステム1は、それぞれのテストデータセット200_iの第1の特徴量の値を訓練済みML-モジュール400の第1の入力ニューロン401に、それぞれのテストデータセット200_iの第2の特徴量の値を訓練済みML-モジュール400の第2の入力ニューロン402に、それぞれのテストデータセット200_iの更なる特徴量の値を訓練済みML-モジュール400の第3の入力ニューロン403に送ってもよい。

30

【0107】

第1のグループのテストデータセット200のうちの一つに対応する出力410の各テスト出力値700_iに依存して、第1の確率が計算され得る。第1の確率は、アプリケーション入力データセットがテストデータセット200の第1のグループに関連付けることができる場合、訓練済みML-モジュール400の出力410のアプリケーション出力値が「1」に等しい確率を示すことができる。アプリケーション入力データセットの更なる特徴量の値が第1のサブセット501の一部である場合、すなわち、更なる特徴量のサブセット閾値より小さい場合、アプリケーション入力データセットは第1のグループに関連付けられてもよい。プロセッサ16は、アプリケーション入力データセットに依存する訓練済みML-モジュール400によって、アプリケーション出力値を計算してもよい。

40

【0108】

プロセッサ16は、「1」に等しく、第1のグループのテストデータセット200に対応するそれらのテスト出力値700(700₁、700₂、700_i、700_m)の数と、第1のグループのテストデータセット200の総数とに100%を乗じた比率として第1の確率を計算してもよい。

【0109】

同様に、コンピュータシステム1は、第2のグループのテストデータセット200_iのうちの一つに対応する各テスト出力値700_iに依存する第2の確率を計算してもよい。

50

第 2 の確率は、アプリケーション入力データセットがテストデータセット 200 の第 2 のグループに関連付けることができる場合、訓練済み ML - モジュール 400 の出力 410 のアプリケーション出力値が「1」に等しい確率を示してもよい。

【0110】

プロセッサ 16 は、「1」に等しく、第 2 のグループのテストデータセット 200 に対応するそれらのテスト出力値 700 (700_1 、 700_2 、 700_i 、 700_m) の数と、第 2 のグループのテストデータセット 200 の総数とに 100% を乗じた比率として第 2 の確率を計算してもよい。

【0111】

バイアスメトリックは、第 1 の確率と第 2 の確率との差に等しくてもよい。バイアスメトリックは、この場合、統計的なパリティ差に対応してもよい。別の例では、バイアスメトリックは、第 1 の確率と第 2 の確率の比に等しくてもよい。バイアスメトリックは、この場合、格差影響比率に対応してもよい。この場合、バイアス制約は、例えば、バイアスメトリックが「0.9」よりも高いことであってもよい。一般に、バイアス制約は、バイアスメトリックが所定のバイアス閾値、例えば「0.9」よりも高い場合、バイアスメトリックがバイアス制約を満たすように設計されてもよい。

10

【0112】

コンピュータシステム 1 は、バイアスメトリックがバイアス制約を満たす場合、例えば、アプリケーション入力データセットに基づく出力 410 のアプリケーション出力値を計算するために、訓練済み ML - モジュールを使用のために解放することができる。

20

【0113】

更なる適用例によれば、コンピュータシステム 1 は、代替的にまたは追加的に、非選択データセットおよび選択されたデータセット、すなわち第 1 のデータセット 10 および第 2 のデータセット 20 に依存する非選択特徴量および選択された特徴量との間の相関の尺度として、第 2 の相関メトリックを計算してもよい。プロセッサ 16 は、選択された特徴量および非選択特徴量の共分散と、選択されたデータセットおよび非選択データセットに依存する、選択された特徴量の標準偏差および非選択特徴量の標準偏差の積との間の比率として、更なるピアソン相関係数を計算するために統計モジュール 53 を実行してもよい。

【0114】

さらに、プロセッサ 16 は、第 2 の相関メトリックが第 2 の閾値、例えば「0.7」よりも大きいかどうかを確認するように構成されてもよい。ML - モジュール 400 の訓練は、非選択データセットの値、すなわち第 2 のデータセット 20 の値 20_i を ML - モジュール 400 の第 2 の入力 402 に送ることを含む上述のように実行されてもよい。

30

【0115】

更なる適用例によれば、プロセッサ 16 は、非選択特徴量の値の第 1 のサブセット 801 および非選択特徴量の値の第 2 のサブセット 802 を定義してもよく、第 1 のサブセット 801 および第 2 のサブセット 802 は不連続のサブセットである。一例では、プロセッサ 16 は、非選択特徴量 810 のサブセット閾値と、データレコード 110 の非選択特徴量の値 20_i とによって、第 1 のサブセット 801 と第 2 のサブセット 802 を定義してもよい。非選択特徴量 810 の閾値より小さい非選択データセット、すなわち図 1 から図 9 で与えられた例による第 2 のデータセット 20 の全ての値は、第 1 のサブセット 801 の一部であってもよく、非選択特徴量 810 の閾値より大きいまたは閾値と等しい非選択データセットの全ての値は、第 2 のサブセット 802 の一部であってもよい。プロセッサ 16 は、第 1 のサブセット 801 および第 2 のサブセット 802 を生成するために、図 8 において矢印 820 によって示される昇順で非選択データセットの値をソートしてもよい。例えば、非選択データセットの値 20_k は第 2 のサブセット 802 の最低値であってもよく、非選択データセットの値 20_{k-1} は第 1 のサブセット 801 の最高値であってもよい。

40

【0116】

図示しない別の例では、プロセッサ 16 は、非選択特徴量 810 のサブセット閾値およ

50

び非選択特徴量の任意生成値によって、第1のサブセット801および第2のサブセット802を定義してもよい。この例によれば、非選択特徴量810の閾値より小さい非選択特徴量の任意生成値の全ての値は、第1のサブセット801の一部であってよく、非選択特徴量810の閾値より大きいまたは閾値と等しい非選択特徴量の任意生成値の全ての値は、第2のサブセット802の一部であってよい。

【0117】

さらに、プロセッサ16は、第2の相関メトリックが第2の閾値よりも大きい場合、非選択特徴量の値の第1のサブセット801または非選択特徴量の値の第2のサブセット802に対するML-モジュール400のバイアスの強さを示す更なるバイアスメトリックを決定してもよい。

10

【0118】

バイアスメトリックの計算と同様に、コンピュータシステム1は、非選択特徴量の値の第1のサブセット801および第2のサブセット802への分割に従って、テストデータセット200をさらに2つのグループに分割してもよい。言い換えれば、非選択特徴量の値のそれぞれのサブセットに対してテストデータセット200の1つの対応するグループが提供されるように、テストデータセット200は分割されてもよい。一例では、テストデータセット200の更なる第1のグループのテストデータセットは、それぞれ、第1のサブセット801に関連付けられる非選択特徴量の値を含んでもよい。同様に、テストデータセット200の更なる第2のグループのテストデータセットは、それぞれ、第2のサブセット802に関連付けられる非選択特徴量の値を含んでもよい。

20

【0119】

更なるバイアスメトリックは、更なるバイアス最適化基準を満たすかどうか、またはどの程度満たすかを確認するために使用されてもよい。更なるバイアス最適化基準は、ユーザによって与えられてもよい。一例では、更なるバイアス最適化基準は、訓練済みML-モジュール400の出力410の更なる所定のテスト出力値を計算するための確率が、非選択特徴量の値から独立していることを含んでもよい。訓練済みML-モジュール400の出力410の更なる所定のテスト出力値は、訓練済みML-モジュールの出力値がバイナリ値である場合、「1」に等しくなってもよい。第1の使用例を参照すると、訓練済みML-モジュール400の出力410の更なる所定のテスト出力値の値「1」は、アラーム信号が設定されている、または設定されるべき状況に対応し得る。

30

【0120】

更なるバイアスメトリックを計算するために、コンピュータシステム1は、訓練済みML-モジュール400によって、および上記のようにそれぞれのテストデータセット200_iに依存して、各テストデータセット200_iに対して、訓練済みML-モジュール400の出力410の対応するテスト出力値700_iを計算してもよい。

【0121】

更なる第1のグループのテストデータセット200のうちの一つに対応する出力410の各テスト出力値700_iに依存して、更なる第1の確率が計算されてもよい。更なる第1の確率は、アプリケーション入力データセットがテストデータセット200の更なる第1のグループに関連付けられる場合、訓練済みML-モジュール400の出力410のアプリケーション出力値が「1」に等しい確率を示すことができる。アプリケーション入力データセットの非選択特徴量の値が第1のサブセット801の一部である場合、すなわち非選択特徴量のサブセット閾値より小さい場合、アプリケーション入力データセットは、更なる第1のグループに関連付けられてもよい。プロセッサ16は、アプリケーション入力データセットに依存する訓練済みML-モジュール400によって、アプリケーション出力値を計算してもよい。

40

【0122】

プロセッサ16は、「1」に等しく、更なる第1のグループのテストデータセット200に対応するそれらのテスト出力値700(700₁、700₂、700_i、700_m)の数と、更なる第1のグループのテストデータセット200の総数とに100%を乗じた

50

比率として更なる第 1 の確率を計算してもよい。

【 0 1 2 3 】

同様に、コンピュータシステム 1 は、更なる第 2 のグループのテストデータセット 2 0 0_i のうちの 1 つに対応する各テスト出力値 7 0 0_i に依存する更なる第 2 の確率を計算してもよい。更なる第 2 の確率は、アプリケーション入力データセットがテストデータセット 2 0 0 の更なる第 2 のグループに関連付けることができる場合、訓練済み ML - モジュール 4 0 0 の出力 4 1 0 のアプリケーション出力値が「 1 」に等しい確率を示すことができる。

【 0 1 2 4 】

プロセッサ 1 6 は、「 1 」に等しく、更なる第 2 のグループのテストデータセット 2 0 0 に対応するそれらのテスト出力値 7 0 0 (7 0 0₁、7 0 0₂、7 0 0_i、7 0 0_m) の数と、更なる第 2 のグループのテストデータセット 2 0 0 の総数とに 1 0 0 % を乗じた比率として更なる第 2 の確率を計算してもよい。

【 0 1 2 5 】

更なるバイアスメトリックは、更なる第 1 の確率と更なる第 2 の確率との間の差に等しくてもよい。別の例では、更なるバイアスメトリックは、更なる第 1 の確率と更なる第 2 の確率の比に等しくてもよい。更なるバイアスメトリックは、この場合、更なる格差影響比率に対応してもよい。この場合、更なるバイアス制約は、例えば、更なるバイアスメトリックが「 0 . 8 5 」よりも高いことであってもよい。一般に、更なるバイアス制約は、更なるバイアスメトリックが所定のバイアス閾値、例えば「 0 . 8 5 」よりも高い場合、更なるバイアス制約を満たすように設計されてもよい。

【 0 1 2 6 】

コンピュータシステム 1 は、更なるバイアスメトリックが更なるバイアス制約を満たす場合、例えば、アプリケーション入力データセットに基づく出力 4 1 0 のアプリケーション出力値を計算するために、訓練済み ML - モジュールを使用のために解放してもよい。

【 0 1 2 7 】

図 9 は、機械学習モジュールを生成するためのコンピュータ実装方法のフローチャートを示す。一例では、機械学習モジュールは、ML - モジュール 4 0 0 であってもよい。

【 0 1 2 8 】

ステップ 1 0 0 1 において、訓練データが提供されてもよい。訓練データは、少なくとも、第 1 の特徴量に対応するデータの第 1 のセットと、第 2 の特徴量に対応するデータの第 2 のセットとを含んでもよい。第 1 の特徴量および第 2 の特徴量のうちの 1 つの特徴量は、選択された特徴量であってもよい。選択された特徴量に対応するデータセットは、選択されたデータセットであってもよい。一例において、訓練データは、訓練データ 1 0 0 であってもよく、コンピュータシステム 1 によって提供されてもよい。

【 0 1 2 9 】

ステップ 1 0 0 2 において、更なるデータセットは、第 1 のデータセットおよび第 2 のデータセットのうちの少なくとも 1 つのデータセットに依存して自動的に生成されてもよい。更なるデータセットは、更なる特徴量に対応してもよい。更なるデータセットは、更なるデータセット 4 0 であってもよい。一例において、更なるデータセット 4 0 のそれぞれの値 4 0_i は、それぞれのデータレコード 1 1 0_i に追加されてもよい。

【 0 1 3 0 】

ステップ 1 0 0 3 において、第 1 の相関メトリックは、少なくとも更なるデータセットと選択されたデータセットとに依存して計算されてもよい。第 1 の相関メトリックは、選択された特徴量と更なる特徴量との間の相関の尺度であってもよい。一例において、第 1 の相関メトリックは、上述のピアソン相関係数であってもよい。

【 0 1 3 1 】

ステップ 1 0 0 4 では、第 1 の相関メトリックが第 1 の閾値より大きいかどうかを確認してもよい。

【 0 1 3 2 】

10

20

30

40

50

ステップ 1005 において、機械学習モジュールの訓練が実行されてもよい。一例において、機械学習モジュールの訓練を実行することは、上述したように訓練モジュール 54 を実行することによって ML - モジュール 400 を訓練することを含んでもよい。

【0133】

ステップ 1006 において、少なくとも異なる特徴量の値の第 1 のサブセットおよび異なる特徴量の値の第 2 のサブセットが定義されてもよい。一例では、第 1 のサブセットは第 1 のサブセット 501 であってもよく、第 2 のサブセットは第 2 のサブセット 502 であってもよい。

【0134】

ステップ 1007 において、第 1 の相関メトリックが第 1 の閾値より大きい場合、バイアスメトリックが決定される。バイアスメトリックは、異なる特徴量の値の第 1 のサブセットまたは異なる特徴量の値の第 2 のサブセットに対する機械学習モジュールのバイアスの強さを示してもよい。一例では、バイアスメトリックは、上述したような格差影響比率であってもよい。

10

【0135】

ステップ 1008 において、バイアスメトリックがバイアス制約を満たすかどうか、例えばプロセッサ 16 によって確認されてもよい。一例では、バイアスメトリックが所定のバイアス閾値よりも大きいかどうかを確認されてもよい。

【0136】

ステップ 1009 において、バイアスメトリックがバイアス制約を満たす場合、訓練済み機械学習モジュールは、使用のために解放されてもよい。

20

【0137】

訓練済み機械学習モジュール、例えば訓練済み ML - モジュール 400 の解放は、訓練済み ML - モジュール 400、例えばモデルパラメータ 430 および訓練済み ML - モジュール 400 の構造パラメータを、ストレージシステム 34 のモデルデータベース 600 に記憶することを含んでもよい。訓練済み ML - モジュール 400 は、モデルパラメータ 430 および構造パラメータ 430 の形態でロードされてもよく、アプリケーション入力データセットに依存する出力変数の上述のアプリケーション値を計算するため、もしくは出力の異なる値を計算するために、またはその両方のために使用されてもよい。コンピュータシステム 1 は、第 1 の相関メトリック、第 2 の相関メトリック、バイアスメトリック、もしくは異なるバイアスメトリック、またはその組み合わせをメタデータの形態でストレージシステム 34 に記憶してもよい。メタデータは、異なる特徴量に対応する少なくとも 1 つの値をそれぞれ含む上記アプリケーション入力データセットに依存する ML - モジュール 400 の出力 410 の上述の異なる値を計算するために訓練済み ML - モジュール 400 を使用するとき、選択された特徴量、もしくは異なる特徴量、またはその両方に関する訓練済み ML - モジュール 400 のバイアスを制御するために使用されてもよい。

30

【0138】

コンピュータ実装方法のステップの番号付けは、これらのステップを実行するための順序を示さない場合がある。例えば、ステップ 1006 は、ステップ 1005 の前に実行されてもよい。

40

【0139】

コンピュータシステム 1 は、ステップ 1005、すなわち ML - モジュール 400 の訓練、およびステップ 1007、すなわちバイアスメトリックの計算のいくつかの繰り返しを実行してもよい。これらの繰り返しのうちの 1 つの繰り返しを実行する前に、プロセッサ 16 は、ML - モジュール 400 の構造の修正を実行してもよい。ML - モジュール 400 の構造の修正は、ML - モジュール 400 がニューラルネットである場合、ニューラルネットの隠れ層の数、もしくは隠れ層におけるニューロンの数、またはその両方など、上述の構造パラメータを変更することを含んでもよい。プロセッサ 16 は、ステップ 1005 およびステップ 1007 の各繰り返しの後に、バイアスメトリックがステップ 100

50

5 およびステップ1007の以前の繰り返しの1つと比較して減少したかどうかを確認してもよい。バイアスメトリックが減少した場合、構造パラメータ430およびモデルパラメータは、ML-モジュール400の最も最近に訓練されたバージョンをML-モジュール400の最良のバージョンとして保存するために、記憶されてもよい。一例では、ステップ1005およびステップ1007のいくつかの繰り返しは、バイアスメトリックが以前に既にバイアス制約を満たしていたとしても、ステップ1008の後に実行されてもよい。ステップ1005およびステップ1007のいくつかの繰り返しを実行することは、ML-モジュール400のバイアスをさらに低減させる可能性がある。この例では、最良のバージョンのML-モジュール400を使用のために解放してもよい。

【0140】

更なる例によれば、ステップ1005およびステップ1007のいくつかの繰り返しと、ステップ1005およびステップ1007の各単一の繰り返しの前に構造パラメータ430を修正することは、バイアスメトリックがバイアス制約を満たした前に実行されてもよい。この例では、ステップ1005およびステップ1007の繰り返しを実行することと、ステップ1005およびステップ1007の各単一の繰り返しの前に構造パラメータ430を修正することは、バイアスメトリックがバイアス制約を満たすようにバイアスメトリックを低減することを可能にし得る。

【0141】

図2は、コンピュータシステム1などのコンピュータシステムが、例えばネットワークアダプタ25を使用して、ネットワーク2000に接続されている例示的なコンピューティング環境を示している。限定されないが、ネットワーク2000は、インターネット、ローカルエリアネットワーク(LAN)、モバイル通信ネットワークなどの無線ネットワークなどの通信ネットワークであってもよい。ネットワーク2000は、クラウドコンピューティングネットワークのようなコンピューティングネットワークを構成してもよい。一例では、ネットワーク2000は、通信システム3000の一部であってもよい。コンピュータシステム1は、ネットワーク2000を介して、通信システム3000のエンティティ300(300₁、300₂、300_i、300_n)との接続を有してもよい。エンティティ300は、それぞれ、ネットワーク2000を介して互いに通信し得るモバイルデバイスであってもよい。コンピュータシステム1は、エンティティ300の1つのそれぞれのエンティティ300_iから1つのそれぞれのデータレコード110_iを受信するように構成されてもよい。それぞれのデータレコード110_iは、1つのそれぞれのエンティティ300_iに対応してもよく、この場合、それぞれのエンティティ300_iのステータスもしくはプロパティまたはその両方を記述してもよい。このように、コンピュータシステム1は、ネットワーク2000から処理されるべき訓練データ100を受信してもよい。それゆえ、訓練データ100は、ネットワーク2000によって提供されてもよい。さらに、コンピュータシステム1は、訓練済みML-モジュール400のモデルパラメータ430の値などの計算結果を、ネットワーク2000を介してコンピュータシステム1に接続された別の計算装置(エンティティ300の1つなど)に提供してもよい。

【0142】

コンピュータシステム1は、ネットワーク2000を介して訓練データ100を受信することに応答して、本明細書に記載された動作を全体的または部分的に実行してもよい。特に、コンピュータシステム1は、ネットワーク2000を介してコンピュータシステム1に接続され得る1つ以上の更なるコンピュータシステムとともに、分散計算でそのような動作を実行してもよい。そのために、コンピュータシステム1もしくは更なる関与するコンピュータシステムまたはその両方は、ネットワーク2000を使用して、専用メモリまたは共有メモリなどの更なる計算リソースにアクセスしてもよい。

【0143】

本発明は、システム、方法、もしくはコンピュータプログラム製品またはその組み合わせであってもよい。コンピュータプログラム製品は、プロセッサに本発明の態様を実行させるためのコンピュータ可読プログラム命令を持つコンピュータ可読記憶媒体(または複数

10

20

30

40

50

の媒体)を含んでよい。

【0144】

コンピュータ可読記憶媒体は、使用するために命令実行装置によって命令を保持および記憶する有形の装置とすることができる。コンピュータ可読記憶媒体は、一例として、電子記憶装置、磁気記憶装置、光記憶装置、電磁記憶装置、半導体記憶装置、または前述を任意に適宜組み合わせたものであってよい。コンピュータ可読記憶媒体のより具体的な一例を列挙すると、ポータブルコンピュータディスク、ハードディスク、ランダムアクセスメモリ(RAM)、読み取り専用メモリ(ROM)、書き込み及び消去可能なランダムアクセスメモリ(EPROMまたはフラッシュメモリ)、静的ランダムアクセスメモリ(SRAM)、ポータブルコンパクトディスク読み取り専用メモリ(CD-ROM)、デジタル多用途ディスク(DVD)、メモリスティック、フロッピーディスク、機械的命令が記録された溝を持つパンチカードまたは隆起構造などの符号化されたデバイス、および前述を任意に適宜組み合わせたものである。本明細書で使用されるコンピュータ可読記憶媒体は、電波または他の自由に伝播する電磁波、導波管または他の伝送媒体を通して伝播する電磁波(例えば、光ファイバケーブルを通過する光パルス)、またはワイヤーを介して送信される電気信号などの一時的な信号自体であると解釈されるべきではない。

10

【0145】

本明細書に記載のコンピュータ可読プログラム命令は、コンピュータ可読記憶媒体からそれぞれのコンピュータ装置/処理装置へダウンロードすることができる。あるいは、ネットワーク(例えばインターネット、LAN、WANもしくはワイヤレスネットワークまたはこれらの組み合わせ)を介して、外部コンピュータまたは外部記憶装置へダウンロード可能である。ネットワークは、銅製伝送ケーブル、光伝送ファイバ、ワイヤレス伝送、ルータ、ファイアウォール、スイッチ、ゲートウェイコンピュータもしくはエッジサーバまたはこれらの組み合わせを備えることができる。各コンピュータ装置/処理装置内のネットワークアダプタカードまたはネットワークインタフェースは、ネットワークからコンピュータ可読プログラム命令を受信し、当該コンピュータ可読プログラム命令を、各々のコンピュータ装置/処理装置におけるコンピュータ可読記憶媒体に記憶するために転送する。

20

【0146】

本発明の動作を実施するためのコンピュータ可読プログラム命令は、アセンブラ命令、命令セットアーキテクチャ(ISA)命令、機械命令、機械依存命令、マイクロコード、ファームウェア命令、状態設定データ、集積回路用構成データ、または、スモールトークやC++などのオブジェクト指向プログラミング言語、および「C」プログラミング言語や類似のプログラミング言語などの手続き型プログラミング言語を含む、1つ以上のプログラミング言語の任意の組み合わせで記述されたソースコードもしくはオブジェクトコードのいずれかとすることができる。コンピュータ可読プログラム命令は、スタンドアロン型ソフトウェアパッケージとして完全にユーザのコンピュータ上で、または部分的にユーザのコンピュータ上で実行可能である。あるいは、部分的にユーザのコンピュータ上でかつ部分的にリモートコンピュータ上で、または、完全にリモートコンピュータもしくはサーバ上で実行可能である。後者の場合、リモートコンピュータは、LANやWANを含む任意の種類ネットワークを介してユーザのコンピュータに接続してもよいし、外部コンピュータに(例えば、インターネットサービスプロバイダを使用してインターネットを介して)接続してもよい。いくつかの実施形態において、例えばプログラマブル論理回路、フィールドプログラマブルゲートアレイ(FPGA)、プログラマブル論理アレイ(PLA)を含む電子回路は、本発明の態様を実行する目的で当該電子回路をカスタマイズするために、コンピュータ可読プログラム命令の状態情報を利用することによって、コンピュータ可読プログラム命令を実行することができる。

30

40

【0147】

本発明の各態様は、本明細書において、本発明の実施形態に係る方法、装置(システム)、およびコンピュータプログラム製品のフローチャートもしくはブロック図またはその

50

両方を参照して説明されている。フローチャートもしくはブロック図またはその両方における各ブロック、および、フローチャートもしくはブロック図またはその両方における複数のブロックの組み合わせは、コンピュータ可読プログラム命令によって実行することができる。

【0148】

上記のコンピュータ可読プログラム命令は、機械を生産するために、プログラマブルデータ処理装置のプロセッサに提供してよい。これにより、コンピュータまたは他のプログラマブルデータ処理装置のプロセッサを介して実行される命令が、フローチャートもしくはブロック図またはその両方における1つ以上のブロックにて特定される機能/動作を実行するための手段を形成する。上記のコンピュータ可読プログラム命令はさらに、コンピュータ、プログラマブルデータ処理装置もしくは他の装置またはこれらの組み合わせに対して特定の態様で機能するよう命令可能なコンピュータ可読記憶媒体に記憶してよい。これにより、命令が記憶された当該コンピュータ可読記憶媒体は、フローチャートもしくはブロック図またはその両方における1つ以上のブロックにて特定される機能/動作の態様を実行する命令を含む製品を構成する。

10

【0149】

また、コンピュータ可読プログラム命令を、コンピュータ、他のプログラマブル装置、または他の装置にロードし、一連の動作ステップを当該コンピュータ、他のプログラマブル装置、または他の装置上で実行することにより、コンピュータ実行プロセスを生成してもよい。これにより、当該コンピュータ、他のプログラマブル装置、または他の装置上で実行される命令が、フローチャートもしくはブロック図またはその両方における1つ以上のブロックにて特定される機能/動作を実行する。

20

【0150】

図面におけるフローチャートおよびブロック図は、本発明の種々の実施形態に係るシステム、方法およびコンピュータプログラム製品の可能な実装形態のアーキテクチャ、機能性、および動作を示している。この点に関して、フローチャートまたはブロック図における各ブロックは、特定の論理機能を実行するための1つ以上の実行可能な命令を含む、命令のモジュール、セグメント、または部分を表すことができる。他の一部の実装形態において、ブロック内に示した機能は、各図に示す順序とは異なる順序で実行してもよい。例えば、連続して示される2つのブロックは、実際には、関係する機能に応じて、略同時に実行してもよいし、または場合により逆順で実行してもよい。なお、ブロック図もしくはフローチャートまたはその両方における各ブロック、および、ブロック図もしくはフローチャートまたはその両方における複数のブロックの組み合わせは、特定の機能または動作を行う専用ハードウェアベースのシステムによって、または専用ハードウェアとコンピュータ命令との組み合わせによって実行することができる。

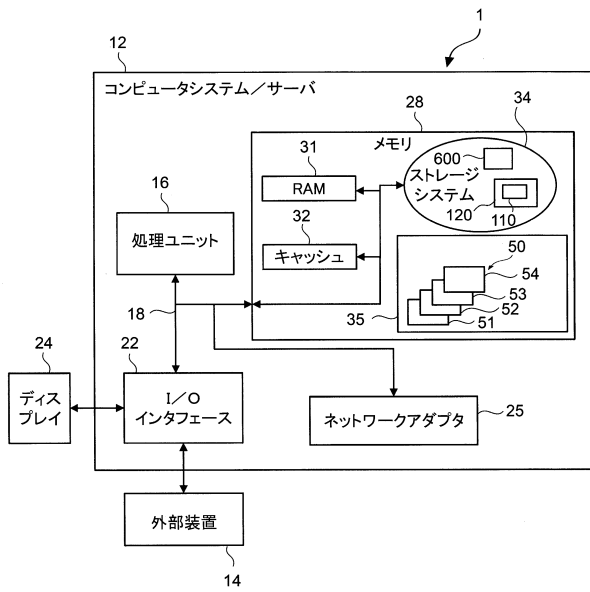
30

40

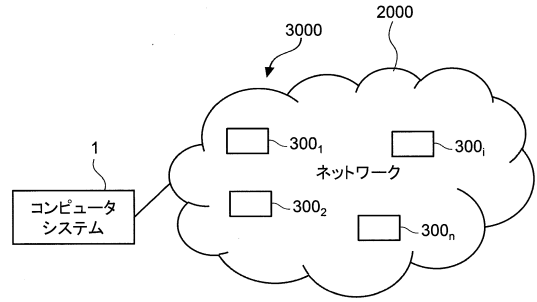
50

【 図 面 】

【 図 1 】



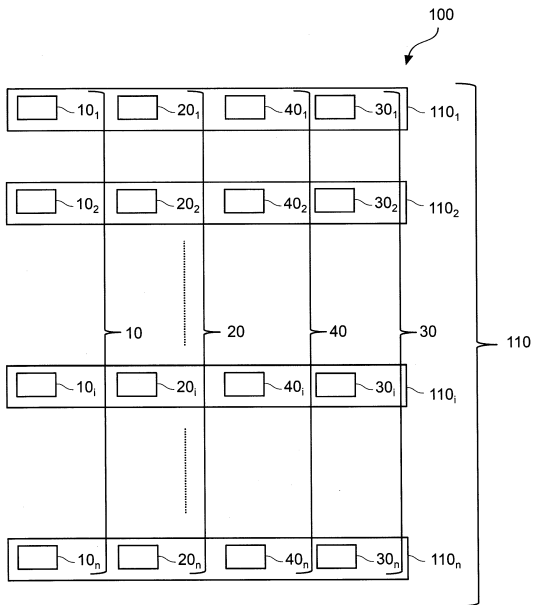
【 図 2 】



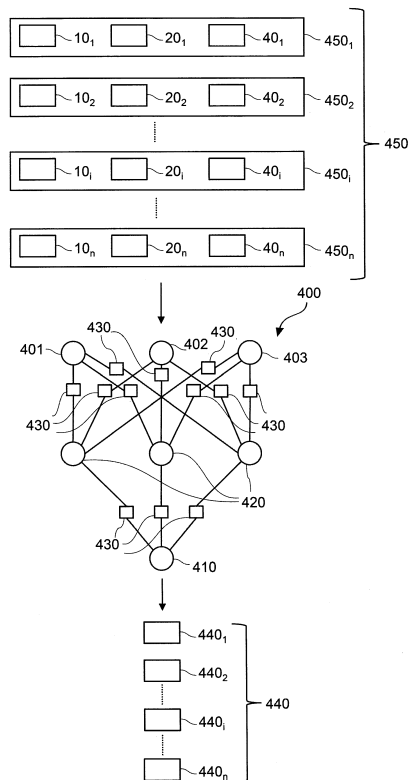
10

20

【 図 3 】



【 図 4 】

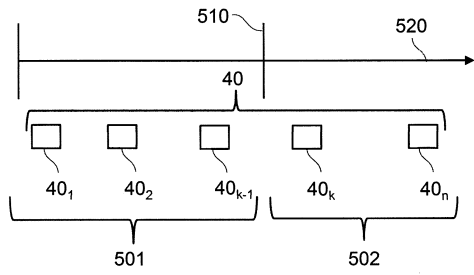


30

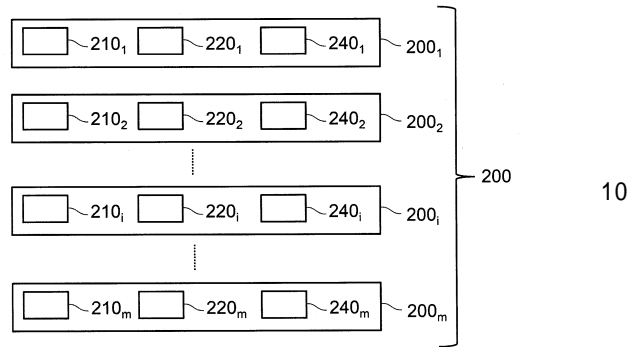
40

50

【 図 5 】



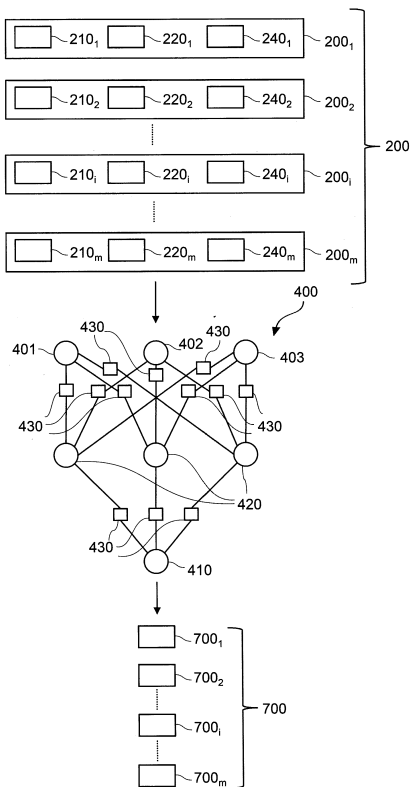
【 図 6 】



10

20

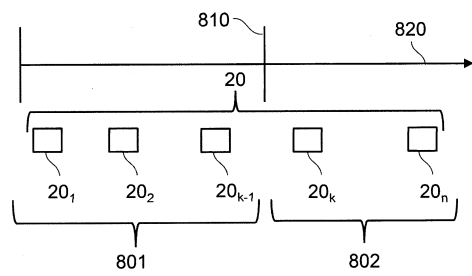
【 図 7 】



30

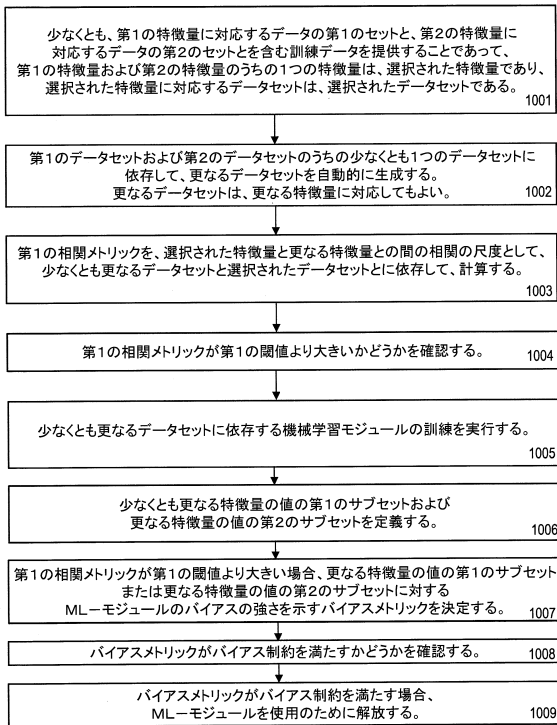
40

【 図 8 】



50

【 図 9 】



10

20

30

40

50

フロントページの続き

(54)【発明の名称】 クグラウンドのバイアスの制御)

(74)代理人 100120710

弁理士 片岡 忠彦

(74)復代理人 100104880

弁理士 古部 次郎

(74)復代理人 100118108

弁理士 久保 洋之

(72)発明者 チミエロフスキ、ジー、ルカシュ

ポーランド共和国 30 - 150 マウオポルスカ県 クラクフ アーミー・クラジョワ 18

(72)発明者 クチャルチク、シモン

ポーランド共和国 30 - 150 マウオポルスカ県 クラクフ アーミー・クラジョワ 18

(72)発明者 ラクザック、ドロタ

ポーランド共和国 30 - 150 マウオポルスカ県 クラクフ アーミー・クラジョワ 18

(72)発明者 リシュカ、ヤクブ、ダニエル

ポーランド共和国 30 - 150 マウオポルスカ県 クラクフ アーミー・クラジョワ 18