IBM SPSS Analytic Server
バージョン 3.1.1

ユーザーズ・ガイド

IBM

注記

本書および本書で紹介する製品をご使用になる前に、 35 ページの『特記事項』に記載されている情報をお読みください。

本書は、IBM SPSS Analytic Server バージョン 3、リリース 1、モディフィケーション 1、および新しい版で明記されていない限り、以降のすべてのリリースおよびモディフィケーションに適用されます。

お客様の環境によっては、資料中の円記号がバックスラッシュと表示されたり、バックスラッシュが円記号と表示されたりする場合があります。

原典： IBM SPSS Analytic Server
Version 3.1.1
User's Guide

発行： 日本アイ・ビー・エム株式会社

担当： トランスレーション・サービス・センター

目次

第 1 章 Analytic Server コンソール

Analytic Server には、データ・ソースおよびプロジェクトを管理するためのシン・クライアント・インターフェースが用意されています。

ログイン

1. ブラウザーのアドレス・バーに Analytic Server の URL を入力します。URL はサーバー管理者から入手できます。
2. サーバーへのログオンに使用するユーザー名を入力します。
3. 指定するユーザー名に関連付けられているパスワードを入力します。

ログインすると、コンソールのホームが表示されます。

コンソールのナビゲート

- ヘッダーには、製品名、現在ログインしているユーザーの名前、およびヘルプ・システムへのリンクが表示されます。現在ログインしているユーザーの名前は、ログアウト・リンクを含むドロップダウン・リストの先頭に表示されます。
- コンテンツ領域には、コンソールのホームから実行できるアクションが表示されます。

データ・ソース

データ・ソースは、レコードのコレクションとデータ・モデルから成り、分析対象のデータ・セットを定義します。レコードのソースには、HDFS 上のファイル (区切り文字で区切られたテキスト・ファイル、固定幅テキスト・ファイル、Excel ファイル)、リレーショナル・データベース、HCatalog、または地理空間を指定できます。データ・モデルは、データの分析に必要なすべてのメタデータ (フィールド名、ストレージ、測定のレベルなど) を定義します。データ・ソースの所有者は、データ・ソースへのアクセスを認可または制限することができます。

データ・ソースのリスト

「データ・ソース (Data sources)」メイン・ページには、現在のユーザーがメンバーになっているデータ・ソースのリストが表示されます。

- データ・ソースの詳細を表示し、プロパティーを編集するには、そのデータ・ソース名をクリックします。
- 検索文字列が名前に含まれるデータ・ソースだけを表示するようにリストをフィルタリングするには、検索領域にその文字列を入力します。
- 「新しいデータ ソースの追加 **(Add new data source)**」ダイアログで指定した名前と内容タイプの新規データ・ソースを作成するには、「新規」をクリックします。
  - データ・ソースに付けることができる名前の制限については、 25 ページの『命名ルール』を参照してください。
  - 選択可能な内容タイプは、ファイル、データベース、および地理空間です。

    注**:**

    - HCatalog オプションを使用できるのは、Analytic Server がこれらのデータ・ソースを処理するように構成されている場合のみです。

- 内容タイプは、一度選択すると編集できません。
- 複数のデータ・ソースを単一のアクションでインポート/エクスポートできます。

- データ・ソースを削除するには、「削除」をクリックします。この操作では、データ・ソースに関連付けられているファイルはすべてそのまま残ります。
- リストを更新するには、「リフレッシュ」をクリックします。
- 「アクション」ドロップダウン・リストを使用すると、選択したアクションが実行されます。
  1. 選択したデータ・ソースからアーカイブを作成してローカル・ファイル・システムにそのアーカイブを保存するには、「エクスポート」を選択します。アーカイブには、選択したデータ・ソースに「プロジェクト」モードまたは「データ・ソース」モードで追加されたファイルがすべて含まれます。

     注**:** 選択したデータ・ソースが 1 つのみのとき、アーカイブ・ファイル名は、選択したデータ・ソース名を共有します。複数のデータ・ソースを選択した場合、アーカイブ・ファイル名は、デフォルトで `datasources.zip` という名前になります。
  2. 「エクスポート」アクションで作成したアーカイブをインポートするには、「インポート」を選択します。

     注**:** 複数のデータ・ソースの情報を含むアーカイブ・ファイルはインポートできません。この場合は、まず個々のデータ・ソースのアーカイブを `datasources.zip` アーカイブから抽出する必要があります。
  3. データ・ソースのコピーを作成するには、「複製 **(Duplicate)**」を選択します。

個別のデータ・ソースの詳細

内容領域は、複数のセクションに分かれています。これらのセクションは、データ・ソースの内容タイプによって異なる場合があります。

詳細　以下の設定はすべての内容タイプに共通です。

名前　データ・ソースの名前を表示する編集可能なテキスト・フィールド。

表示名
: 他のアプリケーションに表示されるデータ・ソースの名前を表示する編集可能なテキスト・フィールド。このフィールドが空白の場合は、表示名として「名前」が使用されます。

説明　データ・ソースに関する説明テキストを指定するための編集可能なテキスト・フィールド。

公開 **(Is public)**
: すべてのユーザーがデータ・ソースを参照できるか (チェック・マークを付けた場合)、ユーザーおよびグループをメンバーとして明示的に追加する必要があるか (チェック・マークを外した場合) を示すチェック・ボックス。

カスタム属性 **(Custom attributes)**
: アプリケーションは、カスタム属性を使用して、データ・ソースにプロパティー (そのデータ・ソースが一時的なものかどうかなど) を追加することができます。これらの属性は、アプリケーションがデータ・ソースをどのように使用しているかについてより詳しい洞察を示すために、Analytic Server コンソールで公開されます。

設定の現在の状態を保持するには、「保存」をクリックします。

共有 **(Sharing)**

以下の設定はすべての内容タイプに共通です。

ユーザーおよびグループを作成者または読者として追加することで、データ・ソースの所有権を共有できます。

- テキスト・ボックスに入力すると、ユーザーおよびグループがフィルタリングされ、名前に検索文字列が含まれるものが表示されます。ドロップダウン・リストから「作成者」または「読者 **(Reader)**」を選択して、データ・ソース内の役割を割り当てます。これらのユーザーおよびグループをメンバーのリストに追加するには、「メンバーを追加 **(Add member)**」をクリックします。
- 参加者を削除するには、メンバー・リストからユーザーまたはグループを選択して、「メンバーの削除 **(Remove member)**」をクリックします。

注**:** 管理者役割を持つユーザーには、メンバーとして明確にリストされているかどうかにかかわらず、すべてのデータ・ソースに対する読み取り権限と書き込み権限があります。

ファイル入力 **(File Input)**

ファイル内容タイプのデータ・ソースを定義する場合に固有の設定。

ファイル・ビューアー **(File Viewer)**

データ・ソースに追加できるファイルが表示されます。「プロジェクト」モードを選択すると、Analytic Server プロジェクト構造内のファイルが表示され、「データ・ソース」を選択すると、データ・ソース内に保管されているファイルが表示され、「ファイル・システム **(File system)**」を選択すると、ファイル・システム (通常は HDFS) が表示されます。どちらのフォルダー構造も参照は可能ですが、HDFS はまったく編集できません。「プロジェクト」モードでは、定義されたプロジェクト内以外ではルート・レベルでのファイルの追加、フォルダーの作成、項目の削除ができません。プロジェクトを作成、編集、または削除するには、「プロジェクト」を使用します。

- 「アップロード **(Upload)**」をクリックすると、現在のデータ・ソースやプロジェクト/サブフォルダーにファイルがアップロードされます。単一ディレクトリー内の複数のファイルを参照して選択することができます。

  注**:** ファイルは、分散ファイル・システムにアップロードされます。アップロードされたファイルは、`/analytic-root` ディレクトリー構造の、該当するテナント、データ・ソース、またはプロジェクト (選択したモードによって異なる)、およびサブフォルダーの下で見つかります。例えば、以下を行った場合:

  1. テナント `ibm` にログオンする。
  2. `fraudDetection` というデータ・ソースを作成する。
  3. 「データ・ソース」モードを選択する。
  4. `historicalData` というサブフォルダーを作成する。
  5. `charges2015.csv` ファイルをアップロードする。

  この結果、そのファイルは、分散ファイル・システム上の `/analytic-root/ibm/.datasource/fraudDetection/historicalData/charges2015.csv` で見つかります。一方で、以下を行った場合:

  1. テナント `ibm` にログオンする。
  2. `fraudDetection` というデータ・ソースを作成する。
  3. 「プロジェクト」モードを選択する。
  4. `creditProcessing` という既存のプロジェクトを選択する。

5. historicalData というサブフォルダーを作成する。
6. charges2015.csv ファイルをアップロードする。

この結果、そのファイルは、分散ファイル・システム上の /analytic-root/ibm/creditProcessing/historicalData/charges2015.csv で見つかります。

- 「新規フォルダー」をクリックすると、「新規フォルダー名 (New Folder Name)」ダイアログで指定した名前の新しいフォルダーが、現在のフォルダーの下に作成されます。
- 「ダウンロード **(Download)**」をクリックすると、選択したファイルがローカル・ファイル・システムにダウンロードされます。
- 「削除」をクリックすると、選択したファイル/フォルダーが削除されます。

データ・ソース定義に含まれるファイル **(Files included in data source definition)**

移動ボタンを使用して、選択したファイルまたはフォルダーをデータ・ソースに追加することや、データ・ソースから削除することができます。データ・ソース内の選択された各ファイルまたはフォルダーについて、「設定」をクリックすると、ファイル読み取りの仕様を定義できます。

データ・ソース内に複数のファイルがある場合、それらのファイルは共通のメタデータを共有する必要があります。つまり、各ファイルが同じ数のフィールドを持ち、それらのフィールドが各ファイル内で同じ順序で解析され、すべてのファイルで各フィールドに同じストレージが割り当てられている必要があります。ファイル間で不一致があると、コンソールが「プレビューとメタデータ (Preview and Metadata)」を作成できないか、Analytic Server がファイルを読み込むときに、有効な値が無効な (NULL) 値として解析される原因となることがあります。

データベース選択 **(Database Selections)**

レコードの内容を含むデータベースに対する接続パラメーターを指定します。

データベース

接続するデータベースの種類を選択します。Db2、Greenplum、Apache Impala、Amazon Redshift、MySQL、Netezza、Oracle、SQL Server、TeraData、Hive、DashDB、または BigSQL から選択してください。目的の種類がリストされていない場合は、適切な JDBC ドライバーでAnalytic Server を構成するように、サーバー管理者に依頼してください。

注**:** Analytic Server では、リモート・システム上にある MySQL データベースがサポートされます。

サーバー・アドレス **(Server address)**

データベースをホストするサーバーの URL を入力します。

サーバー・ポート

データベースが listen するポートの番号。

データベース名

接続するデータベースの名前。

ユーザー名

データベースがパスワードで保護されている場合は、ユーザー名を入力します。

パスワード

データベースがパスワードで保護されている場合は、パスワードを入力します。

テーブル名
使用するデータベースの表の名前を入力します。

最大同時読み取り数 **(Maximum concurrent reads)**
データ・ソースで指定された表からデータを読み込むために、Analytic Server からデータベースに送信することができる同時クエリー数の制限を入力します。

HCatalog の選択 **(HCatalog Selections)**
Apache HCatalog の下で管理されているデータにアクセスするためのパラメーターを指定します。

データベース
HCatalog データベースの名前。

テーブル名
使用するデータベースの表の名前を入力します。

フィルター
表がデータ区分された表として作成されている場合は、表のデータ区分フィルターです。HCatalog のフィルタリングは、文字列型の Hive データ区分キーを使用する場合にのみサポートされています。

注**:** Hadoop の一部ディストリビューションでは、!=、<>、および LIKE の各演算子が機能しないようです。これは、HCatalog とこれらのディストリビューションの間の互換性に関する問題です。

HCatalog のフィールドのマッピング **(HCatalog Field Mappings)**
データ・ソース内のフィールドへの、HCatalog 内の要素のマッピングを表示します。「編集」をクリックすると、フィールドのマッピングを変更できます。

注**:** Hive 表からデータを公開する HCatalog ベースのデータ・ソースを作成すると、その Hive 表が多数のデータ・ファイルで構成されている場合に、Analytic Server がデータ・ソースからデータの読み込みを開始するたびに、大幅な遅延が発生することがあります。データ・ファイルのサイズを増やすことでファイルの数を減らしてから Hive 表を再作成し、ファイル数を 400 未満にしてください。

地理空間の選択 **(Geospatial Selections)**
地理データにアクセスするためのパラメーターを指定します。

地理空間タイプ **(Geospatial type)**
地理データは、オンラインのマップ・サービスやシェープファイルから得ることができます。

マップ・サービスを使用している場合は、そのサービスの URL を指定し、使用するマップ層を選択します。

シェープファイルを使用している場合は、そのシェープファイルを選択するかアップロードします。シェープファイルは実際には、同じディレクトリー内に保存されている、共通のファイル名を持つ一連のファイルである点に注意してください。接尾辞が SHP のファイルを選択します。Analytic Server はその他のファイルを検索して使用します。接尾辞が SHX と DBF の 2 つの追加ファイルが常に存在している必要があります。シェープファイルによっては、いくつかの追加ファイルが存在する場合もあります。

プレビューとメタデータ **(Preview and Metadata)**
データ・ソースの設定を指定したら、「プレビューとメタデータ (Preview and Metadata)」をクリックして、データ・ソースの仕様を確認します。

出力　内容タイプがファイルまたはデータベースであるデータ・ソースには、Analytic Server で実行されるストリームからの出力を付加できます。この付加を有効にするには、「書き込み可能にする **(Make writeable)**」を選択して、以下の手順を実行します。

- 内容タイプがデータベースであるデータ・ソースの場合は、出力データの書き込み先である出力データベース表を選択します。
- 内容タイプがファイルであるデータ・ソースの場合は、以下の手順を実行します。
  1. 新規ファイルの書き込み先である出力フォルダーを選択します。

     ヒント**:** ファイルとデータ・ソースの関連付けを追跡しやすくするために、データ・ソースごとに別々のフォルダーを使用してください。
  2. ファイル形式を選択します。「**CSV**」 (コンマ区切り値) または「分割可能 **2** 進形式 **(Splittable binary format)**」のいずれかを指定します。
  3. オプションで、「シーケンス・ファイルの作成 **(Make sequence file)**」を選択します。これは、下流の MapReduce ジョブで使用できる分割可能圧縮ファイルを作成する場合に使用します。
  4. CSV で出力する場合で、埋め込みの改行や復帰文字を含む文字列フィールドがある場合は、「改行をエスケープ可能 **(Newlines can be escaped)**」 を選択します。これによって、各改行はバックスラッシュに続けて「n」として、復帰はバックスラッシュに続けて「r」として、バックスラッシュは 2 つの連続するバックスラッシュとして記述されます。このようなデータは同一の設定で読み取る必要があります。改行や復帰文字を含む文字列データを処理する際には、分割可能 2 進形式を使用することを強く推奨します。
  5. 圧縮形式を選択します。リストには、Analytic Server のインストール済み環境で使用するように構成されているすべての形式が表示されます。

     注**:** 圧縮形式とファイル形式の組み合わせによっては、分割できない出力が作成されることがあるため、以降に MapReduce 処理を行う場合には不向きです。そのような選択を行うと、Analytic Server の「出力」セクションに警告が表示されます。

設定 (ファイル・データ・ソース)

「設定」ダイアログでは、ファイル・ベース・データの読み取りに関する仕様を定義できます。これらの設定は、選択したすべてのファイルと、選択したフォルダー内のファイルのうち「フォルダー」タブの基準に一致するすべてのファイルに適用されます。

正しくないパーサー設定をファイルに指定すると、コンソールが「プレビューとメタデータ (Preview and Metadata)」を作成できないか、Analytic Server がファイルを読み込むときに、有効な値が無効な (NULL) 値として解析される原因となることがあります。

「設定」タブ

「設定」タブでは、ファイル・タイプと、そのファイル・タイプに固有のパーサー設定を指定できます。

データ・ソースは、サポートされている任意の形式の圧縮ファイルを使用して定義できます。サポートされている圧縮形式は、Gzip、Deflate、Bz2、Snappy、IBM CMX などです。

区切りファイル・タイプ (Delimited file type)

区切りファイルは、フィールドに制限のないテキスト・ファイルです。ファイル内のレコードは一定数のフィールドで構成されますが、各フィールド内の文字数は一定ではありません。区切りファイルのファイル拡

張子は、一般に *.csv または *.tab です。詳細については、『区切りファイル・タイプの設定』を参照してください。

固定長ファイル・タイプ (Fixed file type)

固定長フィールドのテキスト・ファイルは、フィールドが区切り文字で区切られてはいないが、フィールドの開始位置が同じであり、その長さが固定されているファイルです。固定長フィールドのテキスト・ファイルのファイル拡張子は、一般に *.dat です。詳細については、 9 ページの『固定長ファイル・タイプの設定』を参照してください。

準構造化ファイル・タイプ (Semi-structured file type)

準構造化ファイル (*.log など) は、正規表現によってフィールドにマップできる予測可能な構造を持つが、区切りファイルほど厳密に構造化されていないテキスト・ファイルです。詳細については、 10 ページの『準構造化ファイル・タイプの設定』を参照してください。

Text Analytics ファイル・タイプ (Text Analytics file type)

Text Analytics ファイルは、SPSS Text Analytics を使用して分析できるドキュメント (*.doc、*.pdf、*.txt など) です。

空の行をスキップ **(Skip empty lines)**
: 抽出したテキストの内容に含まれる空の行を無視するかどうかを指定します。デフォルトは「いいえ」です。

行区切り文字 **(Line separator)**
: 改行を定義する文字列を指定します。デフォルトは改行文字「¥n」です。

SPSS Statistics ファイル・タイプ (SPSS Statistics file type)

SPSS Statistics ファイル (*.sav、*.zsav) は、データ・モデルを含むバイナリー・ファイルです。このファイル・タイプについては、「設定」タブでさらに設定を行う必要はありません。

分割可能 2 進形式 ファイル・タイプ (Splittable binary format file type)

ファイル・タイプが分割可能 2 進形式ファイル (*.asbf) であることを指定します。このファイル・タイプは、すべての Analytic Server フィールド・タイプを表すことができます (リスト・フィールドをまったく表現できず、埋め込みの改行や復帰を処理するのに特別な設定を必要とする CSV とは異なります) 。このファイル・タイプについては、「設定」タブでさらに設定を行う必要はありません。

シーケンス・ファイル・タイプ (Sequence file type)

シーケンス・ファイル (*.seq) は、Key-Value ペアとして構造化されたテキスト・ファイルです。このファイルは、一般に、MapReduce ジョブで中間形式として使用されます。

Excel ファイル・タイプ (Excel file type)

ファイル・タイプが Microsoft Excel ファイル (*.xls、*.xlsx) であることを指定します。詳細については、 11 ページの『Excel ファイル・タイプの設定』を参照してください。

区切りファイル・タイプの設定:

区切りファイル・タイプには、以下の設定を指定できます。

文字セットのエンコード **(Character set encoding)**
ファイルの文字エンコード。「UTF-8」、「ISO-8859-2」、「GB18030」などの Java 文字セット名を選択または指定します。デフォルトは「**UTF-8**」です。

フィールド区切り文字
フィールドの境界を示す 1 つ以上の文字。それぞれの文字は、個別の区切り文字と解釈されます。例えば、「コンマ」と「タブ」を選択すると (または「その他」を選択して「**,¥t**」と入力すると)、コンマまたはタブがフィールドの境界を示す文字になります。制御文字によってフィールドが区切られる場合は、制御文字に加えて、ここで指定された文字が、区切り文字として扱われます。制御文字によってフィールドが区切られない場合のデフォルトは「,」、それ以外の場合のデフォルトは空の文字列です。

制御文字によるフィールドの区切り **(Control characters delimit fields)**
LF と CR を除くすべての ASCII 制御文字をフィールド区切り文字として扱うかどうかを設定します。デフォルトは「いいえ」です。

最初の行がフィールド名を含む **(First row contains field names)**
最初の行をフィールド名の決定に使用するかどうかを設定します。デフォルトは「いいえ」です。

スキップする先頭文字数 **(Number of initial characters to skip)**
ファイルの先頭でスキップする文字の数。この値は、負でない整数です。デフォルトは 0 です。

空白の結合 **(Merge white space)**
複数のスペースやタブが連続しているときに、それらを単一のフィールド区切り文字と見なすかどうかを設定します。スペースもタブもフィールド区切り文字でない場合は無視されます。デフォルトは「はい」です。

行末コメント文字 **(End-of-line comment characters)**
行末コメントを示す 1 つ以上の文字。レコード上で、この文字以降のものはすべて無視されます。それぞれの文字は、個別のコメント・マーカーと解釈されます。例えば「/*」の場合は、スラッシュまたはアスタリスクのいずれかでコメントが開始すると指定されます。複数の文字からなるコメント・マーカー (「//」など) を定義することはできません。空文字列の場合は、コメント文字を定義しないことを示します。定義した場合は、引用符を処理する前、またはスキップする先頭文字をスキップする前に、コメント文字が検査されます。デフォルトは空文字列です。

不正な文字
無効な文字 (エンコードの文字に対応しないバイト・シーケンス) の処理方法を指定します。

破棄　無効なバイト・シーケンスを破棄します。

置換値
それぞれの無効なバイト・シーケンスを指定された単一の文字で置き換えます。

単一引用符
単一引用符 (アポストロフィ) の処理を指定します。デフォルトは「保存」です。

保存　単一引用符が特別な意味を持たず、他の文字と同じように処理されます。

除去 **(Drop)**
引用符の付いていない単一引用符は削除されます。

ペア **(Pair)**
単一引用符が引用符文字として処理され、単一引用符のペアの間にある文字が特別な意味を持ちません (引用符で囲まれていると見なされます)。単一引用符で囲まれた文字列の中に単一引用符自体を含めることができるかどうかは、設定「二重化によって引用符を引用符で囲むことができる **(Quotes can be quoted by doubling)**」によって決定されます。

二重引用符 **(Double quotation marks)**
二重引用符の処理を指定します。デフォルトは「ペア **(Pair)**」です。

保存　二重引用符が特別な意味を持たず、他の文字と同じように処理されます。

除去 **(Drop)**
引用符の付いていない二重引用符は削除されます。

ペア **(Pair)**
二重引用符が引用文字として処理され、二重引用符のペアの間にある文字が特別な意味を持ちません (引用符で囲まれていると見なされます)。二重引用符で囲まれた文字列の中に二重引用符自体を含めることができるかどうかは、設定「二重化によって引用符を引用符で囲むことができる **(Quotes can be quoted by doubling)**」によって決定されます。

二重化によって引用符を引用符で囲むことができる **(Quotes can be quoted by doubling)**
「ペア **(Pair)**」に設定したときに、二重引用符で囲んだ文字列内で二重引用符を表現できるかどうか、および単一引用符で囲んだ文字列内で単一引用符を表現できるかどうかを示します。「はい」の場合は、二重引用符で囲んだ文字列の内側で二重引用符を 2 個連続して記述するとエスケープされ、単一引用符で囲んだ文字列の内側で単一引用符を 2 個連続して記述するとエスケープされます。「いいえ」の場合は、二重引用符で囲んだ文字列の内側で二重引用符を使用することはできず、単一引用符で囲んだ文字列の内側で単一引用符を使用することもできません。デフォルトは「はい」です。

改行をエスケープ可能 **(Newlines can be escaped)**

パーサーが、バックスラッシュに続けて文字「n」で改行、バックスラッシュに続けて文字「r」で復帰、バックスラッシュに続けてもう 1 つバックスラッシュでバックスラッシュとして解釈するかどうかを示します。改行がエスケープされていない場合は、これらの文字シーケンスは文字通りにバックスラッシュに続いて文字「n」などとして読み取られます。デフォルトは「いいえ」です。

固定長ファイル・タイプの設定**:**

固定長ファイル・タイプには、以下の設定を指定できます。

文字セットのエンコード **(Character set encoding)**
ファイルの文字エンコード。「UTF-8」、「ISO-8859-2」、「GB18030」などの Java 文字セット名を選択または指定します。デフォルトは「**UTF-8**」です。

不正な文字
無効な文字 (エンコードの文字に対応しないバイト・シーケンス) の処理方法を指定します。

破棄　無効なバイト・シーケンスを破棄します。

置換値
それぞれの無効なバイト・シーケンスを指定された単一の文字で置き換えます。

レコード長
レコードの定義方法を示します。「改行区切り **(Newline delimited)**」の場合は、レコードが改行、ファイルの先頭、またはファイルの末尾によって定義され (区切られ) ます。「特定の長さ **(Specific length)**」の場合は、レコードがバイト単位のレコード長によって定義されます。正の値を指定してください。

スキップする先頭レコード数 **(Initial records to skip)**
ファイルの先頭でスキップするレコードの数。負ではない整数を指定します。デフォルト値は 0 です。

フィールド

このセクションはファイル内のフィールドを定義します。「フィールドの追加」をクリックして、フィールド名、フィールド値が始まる列、およびフィールド値の長さを指定します。ファイル内の列には 0 から順に番号が付けられています。

準構造化ファイル・タイプの設定**:**

準構造化ファイルの設定は、ファイルの内容をフィールドにマッピングするためのルールで構成されています。

ルール・テーブル **(Rules Table)**

各ルールは、レコードから情報を抽出して、フィールドを作成します。あわせてルール・テーブルで、データ・ソース内の各レコードから抽出される可能性があるフィールドをすべて定義します。

テーブル内のルールは、各レコードに順に適用されます。テーブル内のすべてのルールがレコードに一致する場合は、他のルール・テーブルを使用してそのレコードを処理する必要がなくなり、次のレコードが処理されます。テーブル内のどのルールも一致しない場合は、テーブル内のそれまでのルールによって抽出されたすべてのフィールド値が破棄されます。別のルール・テーブルがある場合は、そのテーブル内のルールがレコードに適用されます。レコードに一致するテーブルがない場合は、「不一致 (Mismatch)」のルールが適用されます。

不一致 **(Mismatch)**

どのルール・テーブルにも一致しないレコードがある場合は、そのレコードを「スキップ」するか、そのレコード内のすべてのフィールドの値を「欠損」 (NULL) に設定することができます。

ルールのエクスポート **(Export Rules)**

現在表示されているルールを再利用のために保存することができます。エクスポートされたテーブルはサーバー上に保存されます。

ルールのインポート **(Import Rules)**

保存したルール・テーブルを、現在表示されているルール・テーブルにインポートすることができます。この操作を行うと、そのテーブルに定義されているすべてのルールが上書きされるため、新規テーブルを作成してから、ルール・テーブルをインポートすることをお勧めします。

ルール・エディター

ルール・エディターでは、単一のフィールドの抽出ルールを作成できます。

匿名のキャプチャー・グループ **(Anonymous capture group)**

フィールド・キャプチャー・ルールは、通常、前のルールが停止した位置にあるレコードからデータ抽出を開始します。このため、準構造化データ・ソース内の 2 つのフィールドの間に関係のない情報がある場合は、次のフィールドが始まる場所にパーサーを配置する匿名のキャプチャー・グループを定義すると便利です。「匿名のキャプチャー・グループ **(Anonymous capture group)**」を選択すると、キャプチャー・グループに名前やラベルを付けるためのコントロールが無効になりますが、ダイアログの他の部分は通常どおりに機能します。

フィールド名

フィールドの名前を入力します。これは、データ・ソースのメタデータを定義するために使用します。フィールド名は、ルール・テーブル内で固有でなければなりません。

ルール名

オプションで、ルールの説明ラベルを入力します。

説明　オプションで、ルールのより詳しい説明を入力します。

ルールの定義 **(Defining a rule)**

ルールを定義する方法は 2 つあります。

コントロールを使用して抽出ルールを作成する **(Use controls for extraction rules)**

この方法を使用すると、抽出ルールを簡単に作成することができます。

1. フィールド・データの抽出を開始するポイントを指定します。「現在位置 **(Current position)**」を選択すると、前のルールが停止した場所から抽出が開始されます。「次の位置までスキップ **(Skip until)**」を選択すると、レコードの先頭から開始してテキスト・ボックスで指定した文字列に到るまでのすべての文字が無視されます。開始位置にある文字をフィールド・データに含める場合は、「含める **(Include)**」を選択します。
2. 「キャプチャー **(Capture)**」ドロップダウンからフィールド・キャプチャー・グループを選択します。
3. オプションで、フィールド・データの抽出を停止するポイントを選択します。「空白」を選択すると、空白文字 (スペースやタブなど) が出現したときに抽出が停止され、「次の文字 **(At character(s))**」を選択すると、指定した文字列の位置で抽出が停止されます。停止位置にある文字をフィールド・データに含める場合は、「含める **(Include)**」を選択します。

正規表現ルールを手動で定義する **(Manually define regexp rules)**

正規表現構文の作成に抵抗がない場合は、この方法を選択します。「正規表現 **(Regexp)**」テキスト・ボックスに正規表現を入力します。

フィールド・キャプチャー・グループの追加 **(Add Field Capture Group)**

これを選択すると、後から使用するために正規表現を保存できます。保存されたキャプチャー・グループは、「キャプチャー **(Capture)**」ドロップダウンに表示されます。

ルール・エディターには、ルール・テーブル内でこのルールよりも前にあるすべてのルールが適用された後に、このルールによって最初のレコードから抽出されたデータのプレビューが表示されます。

Excel ファイル・タイプの設定**:**

Excel ファイルには、以下の設定を指定できます。

ワークシート選択 **(Worksheet selection)**

データ・ソースとして使用する Excel ワークシートを選択します。数値インデックス (最初のワークシートのインデックスは 0)、またはワークシート名を指定します。デフォルトでは最初のワークシートが使用されます。

インポートするデータ範囲の選択 **(Data range selection for import)**

最初の非ブランク行からデータのインポートを開始するか、明示的なセル範囲を指定してデータをインポートすることができます。

- 「範囲の始点は最初の非ブランク行」。最初の非ブランク・セルを特定し、これをデータ範囲の左上隅として使用します。
- 別の方法として、明示的なセル範囲を行および列で指定することもできます。例えば、Excel 範囲 A1:D5 を指定するには、最初のフィールドに「A1」と入力し、2 番目のフィールドに「D5」 (または「R1C1」、「R5C4」) と入力します。指定した範囲のすべての行 (ブランク行を含む) が返されます。

最初の行がフィールド名を含む **(First row contains field names)**

選択したセル範囲の最初の行にフィールド名を含めるかどうかを指定します。デフォルトは「いいえ」です。

ブランク行の後に読み取りを停止 **(Stop reading after encountering blank rows)**
複数のブランク行が出現した後に、レコードの読み取りを停止するか、またはワークシートの末尾まですべてのデータ (ブランク行を含む) の読み取りを継続するかを指定します。デフォルトは「いいえ」です。

形式

「形式」タブでは、解析されたフィールドのフォーマット設定情報を定義できます。

フィールド変換設定 (Field Conversion Settings)

空白の切り取り **(Trim white space)**
文字列フィールドの先頭および/または末尾から空白文字を削除します。デフォルトは「いいえ」です。以下の値がサポートされています。

なし　空白文字を除去しません。

左　文字列の先頭から空白文字を除去します

右　文字列の末尾から空白文字を除去します

両方　文字列の先頭と末尾から空白文字を除去します。

ロケール
ロケールを定義します。デフォルトはサーバー・ロケールです。ロケール文字列は、<language>[_country[_variant]] の形式で指定する必要があります。ここで、

language
ISO-639 で定義された、小文字 2 文字の有効なコードです。

country
ISO-3166 で定義された、大文字 2 文字の有効なコードです。

variant
ベンダーまたはブラウザー固有のコード。

小数点 **(Decimal separator)**
小数符号として使用する文字を設定します。デフォルトはロケール固有の設定です。

グループ化記号
桁区切り文字に使用されるロケール固有の文字を使用するかどうかを設定します。

デフォルトの日付形式 **(Default date format)**
デフォルトの日付形式を定義します。Unicode Locale Data Markup Language (LDML) 仕様で定義されたすべての形式パターンがサポートされています。

デフォルトの時刻形式 **(Default time format)**
デフォルトの時刻形式を定義します。

デフォルトのタイム・スタンプ **(Default timestamp)**
デフォルトのタイム・スタンプ形式を定義します。

デフォルトのタイム・ゾーン **(Default time zone)**
タイム・ゾーンを設定します。デフォルトは UTC です。この設定は、タイム・ゾーンが明示的に指定されていない時刻フィールドとタイム・スタンプ・フィールドに適用されます。

フィールドのオーバーライド (Field Overrides)

このセクションでは、個別のフィールドにフォーマット設定に関する指示を割り当てることができます。データ・モデルからフィールドを選択するか、フィールド名を入力して、「追加」をクリックすると、個別の

指示を割り当てるフィールドのリストに、そのフィールドが追加されます。フィールドをリストから削除するには、「削除」をクリックします。リストで選択したフィールドに対して、以下のフィールド・プロパティーを設定できます。

ストレージ **(Storage)**
: フィールドのストレージを設定します。

小数点 **(Decimal separator)**
: 実ストレージを持つフィールドに対して、小数符号として使用する文字を設定します。デフォルトはロケール固有の設定です。

グループ化記号
: 整数ストレージまたは実ストレージを持つフィールドに対して、桁区切り文字として使用されるロケール固有の文字を使用するかどうかを設定します。

形式
: 日付、時刻、またはタイム・スタンプの各ストレージを持つフィールドに対して、形式を設定します。形式はドロップダウン・リストから選択します。

「フィールド順序 (Field Order)」タブ

区切りファイル・タイプ、および Excel ファイル・タイプについては、「フィールド順序 (Field Order)」タブで、ファイルのフィールドの解析順序を定義できます。1 つのデータ・ソース内に複数のファイルがある場合は、これが重要になります。フィールドの実際の順序は、ファイル間で異なっている場合がありますが、整合性の取れたデータ・モデルを作成するには、フィールドの解析順序を同じにしておく必要があるからです。

固定長ファイル・タイプ、および準構造化ファイル・タイプについては、この順序が「設定」タブで定義されます。

データ・ソース内のファイルが 1 つだけの場合、またはすべてのファイルでフィールドの順序が同じである場合は、デフォルトの「フィールド順序がデータ・モデルと一致 **(Field order matches data model)**」を使用できます。データ・ソース内に複数のファイルがあり、ファイル内のフィールドの順序が一致しない場合は、ファイルを解析するための「特定のフィールド順序 **(Specific field order)**」を定義します。

1. 順序リストにフィールドを追加するには、フィールド名を入力するか、データ・モデルによって提供されるリストからフィールドを選択します。「すべて追加 **(Add all)**」をクリックすると、データ・モデル内のすべてのフィールドを一度に追加できます。フィールド名は順序リストに一度だけ追加されます。
2. 矢印ボタンを使用して、フィールドを目的の順序に並べ替えます。

「特定のフィールド順序 **(Specific field order)**」を使用する場合、リストに追加されていないフィールドは、このファイルの結果セットに含まれません。データ・モデルに存在するフィールドのうち、このダイアログでリストされていないフィールドがある場合、結果セットではその値が NULL になります。

「フォルダー」タブ

フォルダーに対してパーサー設定を指定するときに、「フォルダー」タブを使用すると、フォルダー内のどのファイルをデータ・ソースに含めるかを選択できます。

選択したフォルダーのすべてのファイルを突き合わせる **(Match all files in the selected folder)**
: データ・ソースに、フォルダーの最上位レベルにあるすべてのファイルを含めます。サブフォルダー内のファイルは含まれません。

正規表現を使用してファイルを突き合わせる **(Match files using a regular expression)**
データ・ソースに、フォルダーの最上位レベルにあるファイルのうち、指定した正規表現に一致するものをすべて含めます。サブフォルダー内のファイルは含まれません。

UNIX ワイルドカード文字式を使用してファイルを突き合わせる **(**再帰的処理の可能性あり**) (Match files using a Unix globbing expression (potentially recursive))**
データ・ソースに、指定した UNIX ワイルドカード文字式に一致するすべてのファイルを含めます。選択したフォルダーのサブフォルダー内のファイルも式に含まれる場合があります。

HCatalog のフィールドのマッピング (HCatalog Field Mappings)

HCatalog スキーマ
指定した表の構造を表示します。HCatalog では、高度に構造化したデータ・セットをサポートできます。そのようなデータに対して Analytic Server データ・ソースを定義するには、構造をフラットにして単純な行と列にする必要があります。スキーマの要素を選択して移動ボタンをクリックすると、その要素を分析のためにフィールドにマップすることができます。

ツリー・ノードのすべてをマップできるわけではありません。例えば、複合タイプの配列またはマップは「親」と見なされるため、直接的にマップできません。HCatalog の配列またはマップ内の単純要素を、1 つずつ個別に追加する必要があります。そのようなノードは、ツリー内でラベルの末尾が ...:array:struct または ...:map:struct であることから識別できます。

以下に例を示します。

- 整数の配列の場合は、配列内の値 (bigintarray[45]) にフィールドを割り当てることはできますが、配列自体 (bigintarray) にフィールドを割り当てることはできません。
- マップの場合は、マップ内の値 (datamap["key"]) にフィールドを割り当てることはできますが、マップ自体 (datamap) にフィールドを割り当てることはできません。
- 整数の配列が配列になっている場合は、値 (bigintarrayarray[45][2]) にフィールドを割り当てることはできますが、配列自体 (bigintarrayarray[45]) にフィールドを割り当てることはできません。

  このため、配列またはマップの要素にフィールドを割り当てるときは、その要素の定義にインデックスまたはキーが含まれている必要があります (bigintarray[index] または bigintmap["key"])。

現在のユーザーが表示できるのは、そのユーザーがアクセスできるテーブルのみです。HDFS ディレクトリーは、読み取り権限と実行権限を持つ唯一のディレクトリーであるため (内部ファイルには読み取り権限があり、このファイルはユーザーが表示できます)、ユーザーは、アクセスできないテーブルを表示できません。この制限は、管理対象 Hive テーブル、外部 Hive テーブル、およびパーティション化されたディレクトリーを保護するために設定されています。

フィールド・マッピング **(Field Mappings)**

HCatalog の要素 **(HCatalog Element)**
編集するにはセルをダブルクリックします。HCatalog の要素が配列またはマップである場合は、セルを編集する必要があります。配列の場合は、フィールドにマップする配列のメンバーに対応する整数を指定します。マップの場合は、フィールドにマップするキーに対応する文字列を引用符で囲んで指定します。

マッピング・フィールド **(Mapping Field)**
Analytic Server データ・ソースに表示されるフィールド。編集するにはセルをダブルクリックします。「マッピング・フィールド (Mapping Field)」列では重複した値は許可されず、エラーとなります。

ストレージ **(Storage)**
フィールドのストレージ。ストレージは HCatalog から取得され、編集できません。

注**:** 「プレビューとメタデータ (Preview and Metadata)」をクリックして HCatalog データ・ソースをファイナライズする場合、編集オプションはありません。

生データ **(Raw Data)**
HCatalog に格納された状態のレコードが表示されます。これは、HCatalog スキーマをフィールドにマップする方法を決定するときに役立ちます。

注**:** 「HCatalog の選択 (HCatalog Selections)」で指定されたフィルター処理が、生データの表示に適用されます。

HCatalog データ・ソースの使用

Analytic Server では HCatalog データ・ソースがサポートされています。このセクションでは、基盤となるさまざまな NoSQL データベースをセットアップする方法について説明します。

通常は、Hive 統合に関するベンダーの資料を参照してください。

Apache Accumulo
https://cwiki.apache.org/confluence/display/Hive/AccumuloIntegration

Apache Cassandra
『Apache Cassandra』

Apache HBase
https://cwiki.apache.org/confluence/display/Hive/HBaseIntegration

MongoDB
https://github.com/mongodb/mongo-hadoop/wiki/Hive-Usage

Oracle NoSQL
https://docs.oracle.com/cd/E57371_01/doc.41/e57351/bigsql.htm#BIGUG21115

XML データ・ソース
17 ページの『XML データ・ソース』

Apache Cassandra

Analytic Server は、Apache Cassandra 内に基となる内容がある HCatalog データ・ソースをサポートしています。

Cassandra は、構造化された Key-Value ストアを提供します。キーは複数の値にマップされ、それが列ファミリーにグループ化されます。列ファミリーはデータベースの作成時に固定されますが、ファミリーへはいつでも列を追加できます。さらに、列は指定されたキーにのみ追加されるため、特定のファミリー内でキーによって列の数が異なる場合があります。各キーの列ファミリーの値は、一緒に保管されます。

Cassandra 表を定義する方法は 2 つあります。従来の Cassandra コマンド行インターフェース (cassandra-cli) を使用する方法と、新しい CQL シェル (csqlsh) を使用する方法です。

Apache Cassandra の外部表が従来の CLI を使用して作成されている場合に、Hive 内にその表を作成するには、以下の構文を使用します。

```
CREATE EXTERNAL TABLE <hive_table_name>  (<column specifications>)
STORED BY 'org.apache.hadoop.hive.cassandra.CassandraStorageHandler'
WITH SERDEPROPERTIES("cassandra.cf.name" = "<cassandra_column_family>",
"cassandra.host"="<cassandra_host>","cassandra.port" = "<cassandra_port>")
TBLPROPERTIES ("cassandra.ks.name" = "<cassandra_keyspace>");
```

例えば、CLI による表定義が以下のようになっているとします。

```
create keyspace test
with placement_strategy = 'org.apache.cassandra.locator.SimpleStrategy'
and strategy_options = [{replication_factor:1}];

create column family users with comparator = UTF8Type;

update column family users with
        column_metadata =
        [
        {column_name: first, validation_class: UTF8Type},
        {column_name: last, validation_class: UTF8Type},
        {column_name: age, validation_class: UTF8Type, index_type: KEYS}
        ];

assume users keys as utf8;

set users['jsmith']['first'] = 'John';
set users['jsmith']['last'] = 'Smith';
set users['jsmith']['age'] = '38';
set users['jdoe']['first'] = 'John';
set users['jdoe']['last'] = 'Dow';
set users['jdoe']['age'] = '42';

get users['jdoe'];
```

この場合、Hive 表の DDL は以下のようになります。

```
CREATE EXTERNAL TABLE cassandra_users  (key string, first string, last string, age string)
STORED BY 'org.apache.hadoop.hive.cassandra.CassandraStorageHandler'
WITH SERDEPROPERTIES("cassandra.cf.name" = "users",
"cassandra.host"="<cassandra_host>","cassandra.port" = "9160")
TBLPROPERTIES ("cassandra.ks.name" = "test");
```

Apache Cassandra の外部表が CQL を使用して作成されている場合に、Hive 内にその表を作成するには、以下の構文を使用します。

```
CREATE EXTERNAL TABLE <hive_table_name>  (<column specifications>)
STORED BY 'org.apache.hadoop.hive.cassandra.cql.CassandraCqlStorageHandler'
WITH SERDEPROPERTIES("cassandra.cf.name" = "<cassandra_column_family>",
"cassandra.host"="<cassandra_host>","cassandra.port" = "<cassandra_port>")
TBLPROPERTIES ("cassandra.ks.name" = "<cassandra_keyspace>");
```

例えば、CQL3 による表定義が以下のようになっているとします。

```
CREATE KEYSPACE TEST WITH REPLICATION = { 'class' : 'SimpleStrategy', 'replication_factor' : 2 };
USE TEST;

CREATE TABLE bankloan_10(
  row int,
  age int,
  ed int,
  employ int,
  address int,
  income int,
  debtinc double,
  creddebt double,
  othdebt double,
  default int,
  PRIMARY KEY(row)
```

```
);

INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (1,41,3,17,12,176,9.3,11.359392,5.008608,1);
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (2,27,1,10,6,31,17.3,1.362202,4.000798,0);
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (3,40,1,15,14,55,5.5,0.856075,2.168925,0);
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (4,41,1,15,14,120,2.9,2.65872,0.82128,0);
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (5,24,2,2,0,28,17.3,1.787436,3.056564,1);
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (6,41,2,5,5,25,10.2,0.3927,2.1573,0);
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (7,39,1,20,9,67,30.6,3.833874,16.668126,0);
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (8,43,1,12,11,38,3.6,0.128592,1.239408,0);
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (9,24,1,3,4,19,24.4,1.358348,3.277652,1);
INSERT INTO bankloan_10 (row, age,ed,employ,address,income,debtinc,creddebt,othdebt,default)
   VALUES (10,36,1,0,13,25,19.7,2.7777,2.1473,0);
```

この場合、Hive 表の DDL は以下のようになります。

```
CREATE EXTERNAL TABLE cassandra_bankloan_10 (row int, age int,ed int,employ int,address int,
   income int,debtinc double,creddebt double,othdebt double,default int)
STORED BY 'org.apache.hadoop.hive.cassandra.cql.CassandraCqlStorageHandler'
WITH SERDEPROPERTIES("cassandra.cf.name" = "bankloan_10","cassandra.host"="<cassandra_host>",
   "cassandra.port" = "9160")
TBLPROPERTIES ("cassandra.ks.name" = "test");
```

XML データ・ソース

Analytic Server では、HCatalog を通じて XML データがサポートされています。

例

1. Hive データ定義言語 (DDL) を通じて、以下のルールに従って、XML スキーマを Hive データ型にマップします。

   ```
   CREATE [EXTERNAL] TABLE <table_name> (<column_specifications>)
   ROW FORMAT SERDE "com.ibm.spss.hive.serde2.xml.XmlSerDe"
   WITH SERDEPROPERTIES (
       ["xml.processor.class"="<xml_processor_class_name>",]
       "column.xpath.<column_name>"="<xpath_query>",
        ...
       ["xml.map.specification.<element_name>"="<map_specification>"
        ...
       ]
   )
   STORED AS
       INPUTFORMAT "com.ibm.spss.hive.serde2.xml.XmlInputFormat"
       OUTPUTFORMAT "org.apache.hadoop.hive.ql.io.IgnoreKeyTextOutputFormat"
   [LOCATION "<data_location>"]
   TBLPROPERTIES (
       "xmlinput.start"="<start_tag ",
       "xmlinput.end"="<end_tag>"
   );
   ```

   注: XML ファイルが Bz2 圧縮を使用して圧縮されている場合は、INPUTFORMAT を com.ibm.spss.hive.serde2.xml.SplittableXmlInputFormat に設定する必要があります。CMX 圧縮を使用して圧縮されている場合は、com.ibm.spss.hive.serde2.xml.CmxXmlInputFormat に設定する必要があります。

   例えば、以下の XML があるとします。

   ```
   <records>
     <record customer_id="0000-JTALA">
       <demographics>
   ```

```
      <gender>F</gender>
      <agecat>1</agecat>
      <edcat>1</edcat>
      <jobcat>2</jobcat>
      <empcat>2</empcat>
      <retire>0</retire>
      <jobsat>1</jobsat>
      <marital>1</marital>
      <spousedcat>1</spousedcat>
      <residecat>4</residecat>
      <homeown>0</homeown>
      <hometype>2</hometype>
      <addresscat>2</addresscat>
    </demographics>
    <financial>
      <income>18</income>
      <creddebt>1.003392</creddebt>
      <othdebt>2.740608</othdebt>
      <default>0</default>
    </financial>
  </record>
</records>
```

これは、以下の Hive DDL で表されます。

```
CREATE TABLE xml_bank(customer_id STRING, demographics map<string,string>, financial map<string,string>)
ROW FORMAT SERDE 'com.ibm.spss.hive.serde2.xml.XmlSerDe'
WITH SERDEPROPERTIES (
    "column.xpath.customer_id"="/record/@customer_id",
    "column.xpath.demographics"="/record/demographics/*",
    "column.xpath.financial"="/record/financial/*"
)
STORED AS
    INPUTFORMAT 'com.ibm.spss.hive.serde2.xml.XmlInputFormat'
    OUTPUTFORMAT 'org.apache.hadoop.hive.ql.io.IgnoreKeyTextOutputFormat'
TBLPROPERTIES (
    "xmlinput.start"="<record customer",
    "xmlinput.end"="</record>"
);
```

詳細については、『XML から Hive のデータ型へのマッピング』を参照してください。

2. Analytic Server コンソールで、HCatalog 内容タイプを使用して、Analytic Server データ・ソースを作成します。

制限

- 現在サポートされているのは XPath 1.0 仕様だけです。
- Hive フィールド名の処理時には、要素および属性の修飾名のローカル部分が使用されます。名前空間プレフィックスは無視されます。

XML から Hive のデータ型へのマッピング: XML でモデル化されたデータを Hive のデータ型に変換するための規則を以下に示します。

構造

XML 要素を Hive の構造タイプに直接マップして、すべての属性をデータ・メンバーにすることができます。要素の内容は、プリミティブ型または複合型の追加のメンバーになります。

XML データ

```
<result name="ID_DATUM">03.06.2009</result>
```

Hive DDL および生データ

```
struct<name:string,result:string>
{"name":"ID_DATUM", "result":"0.3.06.2009"}
```

配列

一連の XML 要素は、Hive のプリミティブ型または複合型の配列として表現できます。以下の例は、XML の <result> 要素の内容を使用して、文字列の配列を定義する方法を示しています。

XML データ

```
<result>03.06.2009</result>
<result>03.06.2010</result>
<result>03.06.2011</result>
```

Hive DDL および生データ

```
result array<string>
```

```
{"result":["03.06.2009","03.06.2010",...]}
```

マップ

XML スキーマでは、マップがネイティブ・サポートされていません。XML でマップをモデル化するための一般的な方法は 3 つあります。各方法に対応するために、以下の構文を使用します。

```
"xml.map.specification.<element_name>"="<key>-><value>"
```

各部の意味は、次のとおりです。

element_name
: マップ項目と見なされる XML 要素の名前

key
: マップ項目キーの XML ノード

value
: マップ項目値の XML ノード

Hive の表作成 DDL では、特定の XML 要素のマップ仕様を、SERDEPROPERTIES セクションで定義する必要があります。キーおよび値は、以下の構文を使用して定義できます。

@attribute
: @attribute を指定すると、属性の値をマップのキーまたは値として使用できます。

element
: 要素名をキーまたは値として使用できます。

#content
: 要素の内容をキーまたは値として使用できます。マップ・キーに使用できるのはプリミティブ型のみであるため、複合型の内容は文字列に変換されます。

XML でマップを表現するための方法、および対応する Hive DDL と生データを、以下に示します。

要素名から内容
: 要素の名前がキーとして使用され、内容が値として使用されます。これは一般的に使用される手法の 1 つであり、XML を Hive のマップ・タイプに関連付けるときにデフォルトで使用されます。この手法では、マップ・キーに指定できるのは文字列型だけであるという制限があることが明らかです。

  XML データ

  ```
  <entry1>value1</entry1>
  <entry2>value2</entry2>
  <entry3>value3</entry3>
  ```

  マッピング、**Hive DDL**、および生データ

  この場合は、デフォルトで要素名がキーとして使用され、内容が値として使用されるため、マッピングを指定する必要はありません。

```
result map<string,string>

{"result":{"entry1": "value1", "entry2": "value2", "entry3": "value3"}}
```

属性から要素の内容

属性値をキーとして使用し、要素の内容を値として使用します。

XML データ

```
<entry name="key1">value1</entry>
<entry name="key2">value2</entry>
<entry name="key3">value3</entry>
```

マッピング、**Hive DDL**、および生データ

```
"xml.map.specification.entry"="@name->#content"

result map<string,string>

{"result":{"key1": "value1", "key2": "value2", "key3": "value3"}}
```

属性から属性

XML データ

```
<entry name="key1" value="value1"/>
<entry name="key2" value="value2"/>
<entry name="key3" value="value3"/>
```

マッピング、**Hive DDL**、および生データ

```
"xml.map.specification.entry"="@name->@value"

result map<string,string>

{"result":{"key1": "value1", "key2": "value2", "key3": "value3"}}
```

複合型の内容

プリミティブ型として使用される複合型の内容は、<string> というルート要素を追加することによって、有効な XML に変換されます。以下の XML を考えてみましょう。

```
<dataset>
 <value>10</value>
 <value>20</value>
 <value>30</value>
</dataset>
```

XPath 式 /dataset/* を実行すると、いくつかの <value> XML ノードが返されます。ターゲット・フィールドがプリミティブ型である場合、実装環境によってクエリー結果に <string> ルート・ノードが追加され、有効な XML に変換されます。

```
<string>
 <value>10</value>
 <value>20</value>
 <value>30</value>
</string>
```

注**:** クエリー結果が単一の XML 要素である場合は、実装環境によってルート要素 <string> が追加されることはありません。

テキストの内容

XML 要素の内容が空白のみのテキストである場合、その内容は無視されます。

プレビューとメタデータ (Preview and Metadata) (データ・ソース)

「プレビューとメタデータ **(Preview and Metadata)**」をクリックすると、データ・ソースのレコードのサンプルとデータ・モデルが表示されます。ここでは、基本的なメタデータ情報を確認できます。

プレビュー

「プレビュー」タブには、レコードの小規模なサンプルと、それらのフィールド値が表示されます。

編集

「編集」タブには、基本的なフィールドのメタデータが表示されます。データ・ソースの内容タイプがファイルの場合は、レコードの小規模なサンプルからデータ・モデルが生成され、このタブでフィールドのメタデータを手動で編集できます。データ・ソースの内容タイプが HCatalog の場合は、HCatalog のフィールド・マッピングに基づいてデータ・モデルが生成され、このタブでフィールド・ストレージを編集することができません。

フィールド

フィールド名をダブルクリックして編集します。

測定 **(Measurement)**

指定されたフィールド内のデータの特性を示す測定の尺度です。

役割 **(Role)**

フィールドが、マシン学習プロセスの入力 (予測フィールド) と対象 (予測済みフィールド) のどちらであるかを、モデル作成ノードに示すために使用されます。使用可能な役割には、「データ区分 (Partition)」の他に、「両方」および「なし」もあります。「データ区分 (Partition)」は、レコードを、学習、テスト、および検証用の個別のサンプルに区分けするために使用されるフィールドを示します。値「分割 (Split)」は、フィールドに入力される可能性のある値ごとに、個別のモデルが作成されるように指定します。「頻度」は、フィールド値が各レコードの頻度の重みとして使用されるように指定します。「レコード ID」は、出力内でレコードを特定するために使用されます。

ストレージ **(Storage)**

ストレージは、データをフィールドに保管する方法を示します。例えば、値 1 および 0 を含むフィールドには、整数データが保管されます。これは、測定の尺度とは異なります。測定の尺度は、データの使用法を示すもので、ストレージには影響しません。例えば、値 1 および 0 を含む整数フィールドの測定の尺度は、フラグ (Flag) に設定できます。これは、通常、1 が True、0 が False であることを示します。

値　「測定」が「カテゴリー」のフィールドの場合は個別の値を示し、「測定」が「連続」のフィールドの場合は値の範囲を示します。

構造　フィールド内のレコードに単一値 (プリミティブ) と値リストのどちらが含まれているかを示します。

深さ **(Depth)**

リストの深さを示します。0 はプリミティブのリスト、1 はリストのリスト、などとなります。

すべてのデータ値をスキャン **(Scan all Data Values)**

これを選択すると、カテゴリー値や範囲制限を判別するためのデータ・ソースのデータ値のスキャンを開始およびキャンセルできます。スキャンが進行中の場合は、「データ・スキャンのキャンセル **(Cancel Data Scan)**」ボタンをクリックします。すべてのデータ値をスキャンすると、メタデータが正しいことを確認できますが、データ・ソースに多数のフィールドやレコードがある場合は時間がかかることがあります。

プロジェクト

プロジェクトは、入力を保管し、ジョブの出力にアクセスするためのワークスペースです。これは、ファイルおよびフォルダーを追加するための最上位の編成構造です。プロジェクトは、個々のユーザーおよびグループと共有することができます。

プロジェクトのリスト

「プロジェクト」メイン・ページには、現在のユーザーがメンバーになっているプロジェクトのリストが表示されます。

- プロジェクトの詳細を表示し、プロパティーを編集するには、そのプロジェクト名をクリックします。
- 検索文字列が名前に含まれるプロジェクトだけを表示するようにリストをフィルタリングするには、検索領域にその文字列を入力します。
- 「新規プロジェクトの追加 **(Add new project)**」ダイアログで指定した名前で新しいプロジェクトを作成するには、「新規」をクリックします。プロジェクトに付けることができる名前の制限については、25 ページの『命名ルール』を参照してください。
- 選択したプロジェクトを削除するには、「削除」をクリックします。このアクションにより、該当するプロジェクトが削除され、そのプロジェクトに関連付けられたすべてのデータが HDFS から削除されます。
- リストを更新するには、「リフレッシュ」をクリックします。

個々のプロジェクトの詳細

内容領域は、「詳細」、「共有 **(Sharing)**」、「ファイル」、および「バージョン」の各省略可能セクションに分かれています。

詳細

名前　プロジェクトの名前を表示する編集可能なテキスト・フィールド。

表示名
: 他のアプリケーションに表示されるプロジェクトの名前を表示する編集可能なテキスト・フィールド。このフィールドが空白の場合は、表示名として「名前」が使用されます。

説明　プロジェクトに関する説明テキストを指定するための編集可能なテキスト・フィールド。

保持するバージョン数 **(Versions to keep)**
: バージョンの数が指定の数を超えた場合に、最も古い確定プロジェクト・バージョンを自動的に削除します。デフォルトは 25 です。

  注**:** クリーンアップ・プロセスはすぐには実行されませんが、20 秒ごとにバックグラウンドで実行されます。

公開 **(Is public)**
: すべてのユーザーがプロジェクトを参照できるか (チェック・マークを付けた場合)、ユーザーおよびグループをメンバーとして明示的に追加する必要があるか (チェック・マークを外した場合) を示すチェック・ボックス。

設定の現在の状態を保持するには、「保存」をクリックします。

共有 **(Sharing)**
: ユーザーおよびグループを作成者または表示者として追加することで、プロジェクトを共有できます。

- テキスト・ボックスに入力すると、ユーザーおよびグループがフィルタリングされ、名前に検索文字列が含まれるものが表示されます。これらのメンバーをメンバー・リストに追加するには、共有のレベルを選択して、「メンバーを追加 **(Add member)**」をクリックします。
  - 作成者はプロジェクトのすべての権限を持つメンバーであり、プロジェクトの他に、その中のフォルダーおよびファイルも変更することができます。これらのユーザー、およびこれらのグループのメンバーは、IBM® SPSS® Modeler を通じて Analytic Server に接続しているときに、このプロジェクトに対する書き込み (Analytic Server エクスポート・ノード) 権限を持ちます。
  - 表示者は、プロジェクト内のフォルダーおよびファイルを表示することができ、プロジェクト内のオブジェクトに対してデータ・ソースを定義することができますが、プロジェクトを変更することはできません。
- 作成者を削除するには、作成者リストでユーザーまたはグループを選択し、「メンバーの削除 **(Remove member)**」をクリックします。

注**:** 管理者は、メンバーとして明示的にリストされているかどうかにかかわらず、すべてのプロジェクトに対する読み取りおよび書き込み権限を持ちます。

注**:** 「共有 (Sharing)」で行われた変更は、すぐに自動的に適用されます。

ファイル

プロジェクト構造ペイン

右側のペインには、現在選択しているプロジェクトのプロジェクト/フォルダー構造が表示されます。フォルダー構造を参照することができますが、ボタン以外の方法で編集することはできません。

- 「ローカル・ファイル・システムへのファイルのダウンロード **(Download file to the local filesystem)**」をクリックすると、選択したファイルがローカル・ファイル・システムにダウンロードされます。
- 「選択したファイルの削除 **(Delete the selected file(s))**」をクリックすると、選択したファイル/フォルダーが削除されます。

ファイル・ビューアー **(File Viewer)**

現在のプロジェクトのフォルダー構造を表示します。フォルダー構造は、定義されたプロジェクト内でのみ編集できます。つまり、「プロジェクト」モードのルート・レベルでは、ファイルの追加、フォルダーの作成、または項目の削除ができません。プロジェクトを作成または削除するには、プロジェクトのリストに戻ります。

- 「**HDFS** にファイルをアップロード **(Upload file to HDFS)**」をクリックすると、現在のプロジェクト/サブフォルダーにファイルがアップロードされます。
- 「新規フォルダーの作成」をクリックすると、「新規フォルダー名 **(New folder name)**」ダイアログで指定した名前の新しいフォルダーが、現在のフォルダーの下に作成されます。
- 「ローカル・ファイル・システムへのファイルのダウンロード **(Download file to the local filesystem)**」をクリックすると、選択したファイルがローカル・ファイル・システムにダウンロードされます。
- 「選択したファイルの削除 **(Delete the selected file(s))**」をクリックすると、選択したファイル/フォルダーが削除されます。

バージョン

プロジェクトは、ファイルおよびフォルダーの内容に加えられた変更に基づいてバージョン管理されます。プロジェクトの属性 (説明など) を変更する場合は、公開されているかどうか、およびどのユーザーと共有しているかにかかわらず、新しいバージョンは不要です。ファイルまたはフォルダーの追加、変更、または削除を行う場合は、新しいバージョンが必要です。

プロジェクトのバージョン管理表

表には、既存のプロジェクト・バージョン、作成日、確定日、各バージョンの担当ユーザー、および親バージョンが表示されます。親バージョンとは、選択したバージョンの基となるバージョンのことです。

- 「ロック」をクリックすると、選択したプロジェクト・バージョンの内容を変更することができます。
- 「確定 **(Commit)**」をクリックすると、プロジェクトに対するすべての変更が保存され、そのバージョンが現行の参照可能状態のプロジェクトになります。
- 「破棄」をクリックすると、ロックしたプロジェクトに対する変更がすべて破棄され、参照可能状態のプロジェクトが最新の確定バージョンに戻ります。
- 「削除」をクリックすると、選択したバージョンが削除されます。

ユーザー管理

管理者は、「ユーザー」ページを通じて、ユーザーおよびグループの役割を管理できます。

内容領域は、「詳細」および「プリンシパル **(Principals)**」の各省略可能セクションに分かれています。

詳細

名前　テナントの名前を表示する、編集できないテキスト・フィールドです。

説明　テナントに関する説明テキストを指定できる編集可能なテキスト・フィールド。

URL　これは、ユーザーが Analytic Server コンソールを通じてテナントにログインできるように、ユーザーに提供する URL です。

状態　「アクティブ **(Active)**」なテナントは現在使用されています。テナントを「非アクティブ **(Inactive)**」にすると、ユーザーがそのテナントにログインすることを防ぎますが、基本情報は削除されません。

プリンシパル **(Principals)**

プリンシパルは、構成時にセットアップされたセキュリティー・プロバイダーから取得されるユーザーおよびグループです。プリンシパルの役割を「管理者」、「ユーザー」、または「読者 (Reader)」に変更できます。

メトリック **(Metrics)**

1 つのテナントに対するリソースの制限を構成できます。テナントが現在使用しているディスク・スペースを報告します。

- テナントの最大ディスク・スペース割り当て量を設定できます。この制限に達すると、テナントのディスク・スペース使用量が割り当て量を下回るために十分なディスク・スペースが消去されるまで、そのテナントでこれ以上のデータをディスクに書き込むことができなくなります。
- テナントのディスク・スペース警告レベルを設定できます。この割り当て量を超えると、テナントのディスク・スペース使用量が割り当て量を下回るために十分なディスク・スペースが消去されるまで、そのテナントでプリンシパルが分析ジョブを実行依頼できなくなります。

- このテナントで一度に実行できる並行ジョブの最大数を設定できます。この割り当て量を超えると、現在実行中のジョブが完了するまで、このテナントでプリンシパルが分析ジョブを実行依頼できなくなります。
- 1 つのデータ・ソースが保持できるフィールドの最大数を設定できます。データ・ソースが作成または更新されるたびに、この制限が確認されます。
- 1 つのデータ・ソースが保持できるレコードの最大数を設定できます。データ・ソースが作成または更新されるたびに、この制限が確認されます。例えば、新しいファイルを作成したときや、ファイルの設定を変更したときです。
- ファイルの最大サイズをメガバイトで設定できます。ファイルがアップロードされる際にこの制限が確認されます。

セキュリティー・プロバイダー構成

ユーザー認証プロバイダーを指定できます。「デフォルト」では、インストールおよび構成時にセットアップされたデフォルトのテナントのプロバイダーが使用されます。「**LDAP**」では、Active Directory や OpenLDAP などの外部 LDAP サーバーを使用してユーザーを認証できます。プロバイダーの設定を指定して、オプションで「プリンシパル」セクションで選択可能なユーザーとグループを制御するためのフィルター設定を指定します。

命名ルール

Analytic Server で固有の名前を付けることができるもの (データ・ソース、プロジェクトなど) については、以下のルールがその名前に適用されます。

- 1 つのテナント内では、同じタイプのオブジェクト内で名前が固有でなければなりません。例えば、2 つのデータ・ソースの両方に insuranceClaims という名前を付けることはできませんが、データ・ソースとプロジェクトに、それぞれ insuranceClaims という名前を付けることはできます。
- 名前には大文字小文字の区別があります。例えば、insuranceClaims と InsuranceClaims は固有の名前と見なされます。
- 名前では、先頭と末尾の空白が無視されます。
- 名前では以下の文字を使用できません。

  `~, #, %, &, *, {, }, ¥¥, :, <, >, ?, /, |, ", ¥t, ¥r, ¥n`

第 2 章 SPSS Modeler の統合

SPSS Modeler は、分析に対する視覚的なアプローチを備えたデータ・マイニング・ワークベンチです。ジョブに含まれる個別のアクションは、データ・ソースへのアクセスから、レコードの結合、新規ファイルの記述、またはモデルの作成に至るまで、すべてキャンバス上のノードとして表されます。これらのアクションをリンクすることで、分析ストリームを形成します。Analytic Server と共に実行する分析ストリームを作成するには、以下のようにします。

1. このストリームは Analytic Server ソース・ノードで開始する必要があります。
2. 通常どおり Modeler インターフェースでストリームの中間を作成します。その際に、 Analytic Server でサポートされるプロセス・ノード (フィールド操作またはレコード操作) を選択します。サポート対象のノードを表示する Analytic Server パネルが Modeler パレットにあります。
3. ストリームを終了するためのいくつかのオプションがあります。
   - Analytic Server によってサポートされるターミナル・ノード (出力、グラフ作成、エクスポート、またはモデル作成) を選択します。この場合、Modeler はストリーム全体を Analytic Server にプッシュします。Analytic Server は、Hadoop クラスターで必要なジョブを調整し、Modeler がその結果を使用できるようにします。 Modeler は、ストリームがローカルで処理される場合と同様に、その結果を取得してユーザーに示します。
   - Analytic Server でサポートされないターミナル・ノードをユーザーが選択した場合、Modelerは、可能な限り多くのストリームを Analytic Server にプッシュし、続いて Hadoop からのレコードのプルを開始します。現在 Analytic Server で作成できないモデルは、Analytic Server でスコアリングできることに注意してください。これは、ストリームを構造化して、Analytic Server でビッグデータの統計的に有効なサブサンプルを取得し、Modeler で「ローカルに」モデルを作成できることを意味します。その後、結果としてのモデル・ナゲットを、完全に Analytic Server で実行されるスコアリング・ストリームに含めることができます。

   注: SPSS Modeler が Hadoop からダウンロードするレコードの最大数は、Analytic Server ストリームのプロパティーで設定できます。

サポート対象ノード

多くの SPSS Modeler ノードでは、HDFS での実行がサポートされていますが、実行方法に相違点があるノードや、現時点でサポートされていないノードもあります。このトピックでは、現在のサポート・レベルについて詳しく説明します。

注: これらのノードの通常の操作については、「SPSS Modelerの資料」を参照してください。

全般

- ここで示された Modeler のフィールド名では、通常どおりに受け入れ可能な文字の一部が、Analytic Server では受け入れられません。
- Modeler ストリームを Analytic Server で実行するには、ストリームを 1 つ以上の Analytic Server ソース・ノードで開始し、単一のモデル作成ノードまたは Analytic Server エクスポート・ノードで終了する必要があります。
- 連続型対象のストレージは、整数ではなく実数に設定することをお勧めします。スコアリング・モデルでは、連続型対象の出力データ・ファイルに必ず実数値が書き込まれるのに対し、スコアの出力データ・モデルでは、対象のストレージに従って処理が行われます。このため、連続型対

象のストレージが整数である場合は、書き込み値とスコアのデータ・モデルに不一致が生じ、この不一致によって、スコアリングされたデータの読み取り時にエラーが発生します。

入力

- Analytic Server ソース・ノード以外のノードで開始されるストリームは、ローカルで実行されます。

レコード操作

ストリーミング時系列分析ノードとスペース タイム ボックス・ノードを除くすべてのレコード操作がサポートされています。以下では、サポートされるノード機能について、特に注意が必要な点を示します。

条件抽出

- フィールド作成ノードがサポートする機能と同じ一連の機能をサポートします。

サンプル

- ブロック・レベルのサンプリングはサポートされていません。
- 複雑なサンプリング方法はサポートされていません。
- **「サンプルを破棄」**を指定したときの最初の n 件のサンプリングはサポートされていません。
- N>20000 を指定したときの最初の n 件のサンプリングはサポートされていません。
- **「最大サンプル数」**が設定されていないときは、n 件ごとのサンプリングはサポートされていません。
- N * **「最大サンプル数」** > 20000 のときは、n 件ごとのサンプリングはサポートされていません。
- ランダム % のブロック・レベルのサンプリングはサポートされていません。
- ランダム % は現在、シードの提供をサポートしています。

レコード集計

- 連続キーはサポートされていません。データをソートするように設定され、この設定をレコード集計ノードで使用する既存のストリームを再利用している場合、ソート・ノードを削除するようにそのストリームを変更してください。
- 順序統計量 (中央値、第 1 四分位数、第 3 四分位数) は概算値が算出され、「最適化」タブを通じてサポートされています。

ソート

- 「最適化」タブはサポートされていません。

分散環境では、ソート・ノードによって設定されたレコード順序を保持する操作の数が限られます。

- ソート・ノードの後にエクスポート・ノードを使用すると、ソートされたデータ・ソースが生成されます。
- レコードのサンプリングが「初めの **n** 件」であるサンプル・ノードをソート・ノードの後に使用すると、先頭から *N* 件のレコードが返されます。

一般にソート・ノードは、ソートされたレコードが必要になる操作のできるだけ近くに配置してください。

レコード結合

- 順序による結合はサポートされていません。

- 「最適化」タブはサポートされていません。
- 結合操作は比較的低速です。HDFS 内に使用可能なスペースがある場合は、各ストリームでデータ・ソースを結合するよりも、一度データ・ソースを結合してから、結合されたソースを以降のストリームで使用する方が、処理速度が大幅に向上する可能性があります。

R 変換

ノードの R シンタックスは、レコード単位の操作で構成されている必要があります。

フィールド操作

匿名化、行列入替、時間区分、および履歴の各ノードを除くすべてのフィールド操作がサポートされています。以下では、サポートされるノード機能について、特に注意が必要な点を示します。

自動データ準備

- ノードの学習はサポートされていません。学習した自動データ準備ノードの変換の新規データへの適用はサポートされています。

フィールド作成

- 順序機能を除くすべてのフィールド作成機能がサポートされています。
- 新しいフィールドの度数としての作成は、本質的にシーケンス操作であるため、サポートされていません。
- 分割フィールドは、分割として使用されている同じストリーム内では作成できません。分割フィールドを作成するストリームと、フィールドを分割として使用するストリームの 2 つのストリームを作成する必要があります。

置換

- フィールド作成ノードがサポートする機能と同じ一連の機能をサポートします。

データ分割

以下の機能はサポートされていません。

- 最適なデータ分割
- ランク
- 分位 -> 分位: 値の合計
- 分位 -> 同順位: 現在のまま保持および無作為割当
- 分位 -> カスタム N: 100 を超える値と、100 % N が 0 に等しくない任意の N 値。

RFM 分析

- 同順位の処理方法としての「現在のまま保持」オプションはサポートされません。RFM のリーセンシー、度数、およびマネタリーの各スコアは、同じデータから Modeler によって計算されたスコアと一致するとは限りません。スコアの範囲は同じですが、スコアの割り当て (ビン数) がそれぞれ異なっている場合があります。

グラフ作成

すべてのグラフ作成ノードがサポートされています。

モデル作成

サポートされているモデル作成ノードは、時系列、TCM、ツリー-AS、C&R ツリー、QUEST、CHAID、線型、線型-AS、ニューラル・ネット、GLE、LSVM、TwoStep-AS、ランダム・ツリー、STP、およびアソシエーション・ルールです。以下では、これらのノードの機能について、特に注意が必要な点を示します。

線形　ビッグデータに対するモデルを作成する場合は、通常、目的を「特に大きなデータセット」に変更するか、分割を指定してください。

- 既存の PSM モデルの継続学習はサポートされていません。
- 標準のモデル作成目的は、各分割のレコード数が多くなりすぎないように分割フィールドを定義する場合にのみお勧めします。ここで、「多すぎる」の定義は、ご使用の Hadoop クラスターの各ノードの処理能力によって異なります。一方、モデルを作成するためのレコードが少なくなりすぎることを防ぐため、分割をあまり細かく定義しないように注意する必要があります。
- 「ブースティング」目的はサポートされていません。
- 「バグ」目的はサポートされていません。
- 「特に大きいデータ・セット」目的は、レコードが少数の場合はお勧めしません。これは、モデルが作成されないか、品質の劣るモデルが作成されることが多いためです。
- 自動データ準備はサポートされていません。このため、欠損値の多いデータに基づいてモデルを作成しようとしたときに、問題が発生する可能性があります。これらの値は、通常、自動データ準備の一環として代入されるからです。回避策として、ツリー・モデルまたはニューラル・ネットワークを拡張設定で使用して、選択した欠損値を代入することができます。
- 精度統計は、分割モデルについては計算されません。

ニューラル・ネットワーク

ビッグデータに対するモデルを作成する場合は、通常、目的を「特に大きなデータセット」に変更するか、分割を指定してください。

- 既存の標準または PSM モデルの継続学習はサポートされていません。
- 標準のモデル作成目的は、各分割のレコード数が多くなりすぎないように分割フィールドを定義する場合にのみお勧めします。ここで、「多すぎる」の定義は、ご使用の Hadoop クラスターの各ノードの処理能力によって異なります。一方、モデルを作成するためのレコードが少なくなりすぎることを防ぐため、分割をあまり細かく定義しないように注意する必要があります。
- 「ブースティング」目的はサポートされていません。
- 「バグ」目的はサポートされていません。
- 「特に大きいデータ・セット」目的は、レコードが少数の場合はお勧めしません。これは、モデルが作成されないか、品質の劣るモデルが作成されることが多いためです。
- データ内に多くの欠損値がある場合は、拡張設定を使用して欠損値を代入してください。
- 精度統計は、分割モデルについては計算されません。

C&R ツリー、**CHAID**、および **QUEST**

ビッグデータに対するモデルを作成する場合は、通常、目的を「特に大きなデータセット」に変更するか、分割を指定してください。

- 既存の PSM モデルの継続学習はサポートされていません。
- 標準のモデル作成目的は、各分割のレコード数が多くなりすぎないように分割フィールドを定義する場合にのみお勧めします。ここで、「多すぎる」の定義は、ご使用の Hadoop クラスターの各ノードの処理能力によって異なります。一方、モデルを作成するためのレコードが少なくなりすぎることを防ぐため、分割をあまり細かく定義しないように注意する必要があります。
- 「ブースティング」目的はサポートされていません。

- 「バグ」目的はサポートされていません。
- 「特に大きいデータ・セット」目的は、レコードが少数の場合はお勧めしません。これは、モデルが作成されないか、品質の劣るモデルが作成されることが多いためです。
- インタラクティブ・セッションはサポートされていません。
- 精度統計は、分割モデルについては計算されません。
- 分割フィールドが存在する場合、Modeler のローカルに作成されたツリー・モデルは Analytic Server が作成したツリー・モデルと少し異なるため、異なるスコアが生成されます。どちらの場合のアルゴリズムも有効です。Analytic Server が使用するアルゴリズムの方が単に新しいだけです。ツリー・アルゴリズムは多くのヒューリスティック・ルールを持つ傾向にあるという事実を踏まえると、2 つのコンポーネント間の違いは正常です。

モデルのスコアリング

モデル化がサポートされているすべてのモデルでは、スコアリングもサポートされています。さらに、一部のノードのローカルで作成されたモデル・ナゲットでも、スコアリングがサポートされています。そのノードとは、C&RT、Quest、CHAID、線型、ニューラル・ネット (モデルが標準、ブースティング、バギングのいずれか、または非常に大きなデータ・セット用であるかどうかにかかわらず)、回帰、C5.0、ロジスティック、一般化線型、GLMM、Cox、SVM、ベイズ・ネット、TwoStep、KNN、ディシジョン・リスト、判別、自己学習、異常値検出、Apriori、Carma、K-Means、Kohonen、R、およびテキスト・マイニングです。

- 未加工または調整された傾向はスコアリングされません。回避策として、フィールド作成ノードで以下の式を使用して、未加工の傾向を手動で計算することによって、同じ効果を得られる場合があります: if 'predicted-value' == 'value-of-interest' then 'prob-of-that-value' else 1-'prob-of-that-value' endif

R ナゲットの R シンタックスは、レコード単位の操作で構成されている必要があります。

出力 クロス集計、精度分析、データ検査、変換、グローバルの設定、記述統計、平均、およびテーブルの各ノードがサポートされています。以下では、サポートされるノード機能について、特に注意が必要な点を示します。

データ検査

データ検査ノードは連続型フィールドのモードを生成できません。

平均 平均ノードは標準誤差や 95% 信頼区間を生成できません。

テーブル

テーブル・ノードは、上流の操作の結果を含む一時的な Analytic Server データ・ソースを記述することによってサポートされます。テーブル・ノードは、その後、そのデータ・ソースの内容をページングします。

エクスポート

ストリームを、Analytic Server ソース・ノードで開始して、Analytic Server エクスポート・ノード以外のエクスポート・ノードで終了することができますが、データは HDFS から SPSS Modeler Server に移動され、最終的にエクスポート場所に移動されます。

ベスト・プラクティス

HCatalog/Hive へのプッシュバック

データ区分された Hive 表のデータを処理する場合、必要な区分の選択を Hive にプッシュバックするための Modeler ストリームを構成できます。

1. HCatalog/Hive データ・ソースを参照する Analytic Server ソース・ノードからストリームを開始します。
2. Hive 表のデータ区分フィールドとして使用されるフィールド専用のレコードを選択する、条件抽出ノードに接続します。データ区分フィールドとして使用されないフィールドが条件抽出ノードの式で参照されている場合、ストリームは HCatalog/Hive にプッシュバックしません。
3. 通常どおりに他のノードに接続します。

第 3 章 トラブルシューティング

このセクションでは、いくつかの一般的な使用上の問題とそれらの修正方法について説明します。

データ・ソース

HCatalog データ・ソースのパーティション化された列に定義されたフィルターが履行されない
これは、一部のバージョンの Hive で見られる問題であり、以下のシチュエーションで見られます。

- HCatalog データ・ソースを定義し、そのデータ・ソース定義にフィルターを定義する場合。
- パーティション化されたテーブル列を参照するフィルター・ノードを持つ Modeler ストリームを作成する場合。

回避策は、パーティション化された列と等しい値を持つ新規フィールドを作成するフィールド作成ノードを Modeler ストリームに追加することです。フィルター・ノードはこの新規フィールドを参照する必要があります。

Oracle NoSQL

Oracle NoSQL データ・ソースに接続するときに「実行に失敗しました」エラーが検出される
この問題は、HiveKVStorageHandler.jar ストレージ・ハンドラーの期限が切れた結果として発生します。更新されたストレージ・ハンドラーを使用する必要があります。更新されたファイルは、https://github.com/dvasilen/HiveKVStorageHandler3/raw/HADOOP_2.6-HIVE-1.2.0-KV-3.3.4/release/hive-kv-storage-handler-1.2.0-3.3.4.jar にあります。

`hive-kv-storage-handler-1.2.0-3.3.4.jar`

1. JAR ファイルを、Hive の `{HIVE_HOME}/auxlib` ディレクトリーと、Analytic Server の `{AS_ROOT}/ae_wlpserver/usr/servers/aeserver/apps/AE_BOOT.war/WEB-INF/lib` ディレクトリーにコピーします。
2. `{AS_ROOT}/bin/hdfsUpdate.sh` を実行して、変更内容を HDFS に伝搬します。
3. Analytic Server を再始動して、変更内容を反映させます。

注**:** Oracle NoSQL 3.0 データベースを使用するときは、
`oracle.kv.hadoop.hive.table.TableStorageHandler` ストレージ・ハンドラー・クラスをお勧めします。このクラスは、ユーザーが表メタフォーを使用してデータを整理することを必要とします。

特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものです。本書は IBM から他の言語で入手できる場合があります。ただし、これを入手するには、本製品または当該言語版製品を所有している必要がある場合があります。

本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒103-8510
東京都中央区日本橋箱崎町19番21号
日本アイ・ビー・エム株式会社
法務・知的財産
知的財産権ライセンス渉外

IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態で提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

記載されている性能データとお客様事例は、例として示す目的でのみ提供されています。実際の結果は特定の構成や稼働条件によって異なります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確証できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者にお願いします。

IBM の将来の方向性および指針に関する記述は、予告なく変更または撤回される場合があります。これらは目標および目的を提示するものにすぎません。

表示されている IBM の価格は IBM が小売り価格として提示しているもので、現行価格であり、通知なしに変更されるものです。卸価格は、異なる場合があります。

本書はプランニング目的としてのみ記述されています。記述内容は製品が使用可能になる前に変更になる場合があります。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、類似する個人や企業が実在しているとしても、それは偶然にすぎません。

著作権使用許諾:

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、類似する個人や企業が実在しているとしても、それは偶然にすぎません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生的創作物にも、次のように、著作権表示を入れていただく必要があります。

© (お客様の会社名) (西暦年). このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。

商標

IBM、IBM ロゴおよび ibm.com は、世界の多くの国で登録された International Business Machines Corporation の商標です。他の製品名およびサービス名等は、それぞれ IBM または各社の商標である場合があります。現時点での IBM の商標リストについては、http://www.ibm.com/legal/copytrade.shtml をご覧ください。

Adobe、Adobe ロゴ、PostScript、PostScript ロゴは、Adobe Systems Incorporated の米国およびその他の国における登録商標または商標です。

IT Infrastructure Library は AXELOS Limited の登録商標です。

インテル、Intel、Intel ロゴ、Intel Inside、Intel Inside ロゴ、Centrino、Intel Centrino ロゴ、Celeron、Xeon、Intel SpeedStep、Itanium、および Pentium は、Intel Corporation または子会社の米国およびその他の国における商標または登録商標です。

Linux は、Linus Torvalds の米国およびその他の国における登録商標です。

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

ITIL は AXELOS Limited の登録商標です。

UNIX は The Open Group の米国およびその他の国における登録商標です。

Cell Broadband Engine は、Sony Computer Entertainment, Inc.の米国およびその他の国における商標であり、同社の許諾を受けて使用しています。

Linear Tape-Open、LTO、LTO ロゴ、Ultrium および Ultrium ロゴは、HP、IBM Corp. および Quantum の米国およびその他の国における商標です。

Printed in Japan

日本アイ・ビー・エム株式会社
〒103-8510 東京都中央区日本橋箱崎町19-21