RapidMiner で始める簡単データ分析

はじめに

RapidMiner をダウンロードしてみたもののどのように使えばよいか分からないという方に向けた、使い方の超入門講座です。RapidMiner Studio の基本機能と特徴について、チュートリアルを通して学んでいただき、RapidMiner の操作に慣れていただきます。全て読み終えた頃には、皆さまご自身でデータ分析ができるようになります。

RapidMiner とは

> 現代の企業ビジネスに合った分析プラットフォームです。RapidMiner を使えば、「プロセス」と呼ばれる分析処理フローを速く、簡単に作成できます。そして、プロセスに含まれる「オペレータ」と呼ばれるブロックを繋げていけば、予測モデルを作成できます。

それでは、使い方入門講座を始めていきましょう！
※　RapidMiner のダウンロードはこちら

チュートリアル概要

Part1：データ読み込み
RapidMiner Studio の環境設定とデータを取り込む方法
Part2：データ可視化
より良い知見を得るためのデータの可視化方法
Part3：モデル作成
離反しそうな顧客を予測するモデルを作成する方法
Part4：モデル適用
（Part3 とは別の）データにモデルを当てはめる方法
Part5：モデル評価
良い結果を保証するモデルの精度を評価する方法

ビジネスの例を挙げて考えてみましょう。

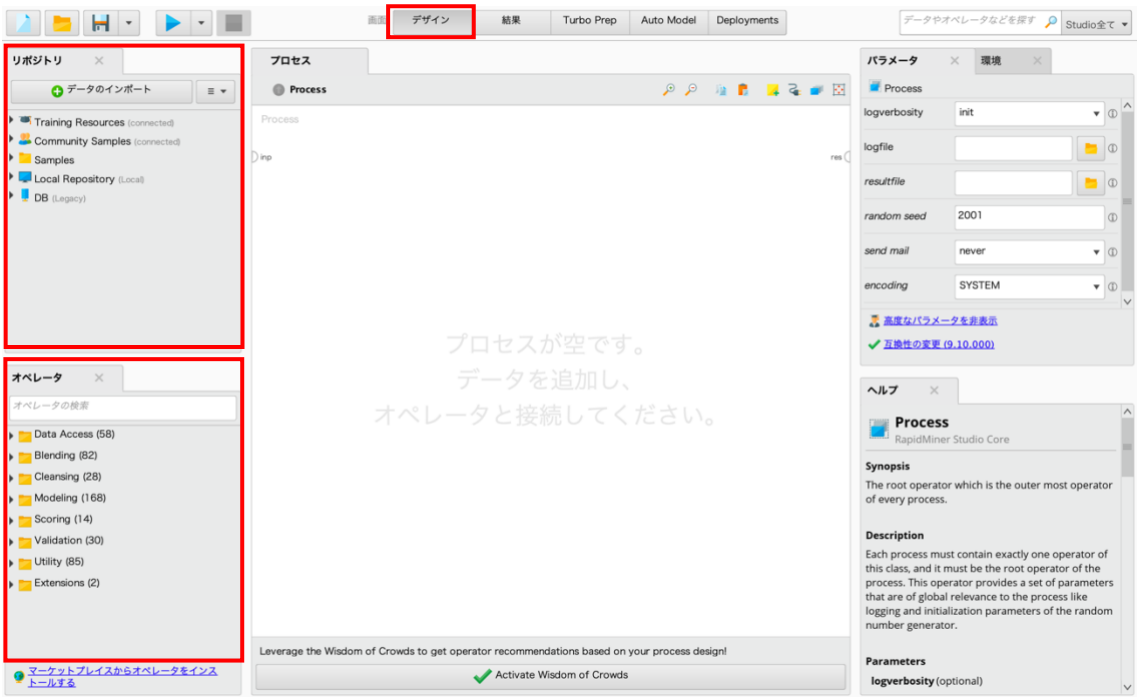
あなたはこれまでに、顧客に関する膨大なデータを収集・蓄積してきました。このデータを使って、各顧客が今後、継続顧客のままか離反顧客になりそうかを見極めたいと考えています。どの顧客が離反するか、どのようにして予測すればよいでしょうか？また、継続顧客として留め置くためには、どの顧客にマーケティング費用を費やせばよいでしょうか？

Part1： データ読み込み

チュートリアルの目標は、顧客が離反するか継続するかを、RapidMiner を使って予測することです。顧客が離反するか継続するか、どちらのカテゴリに属するかを予測するプロセスは、「分類」と呼ばれます。顧客を分類するためにはまず、データを読み込むことから始める必要があります。

データの読み込みは以下の手順で行います。

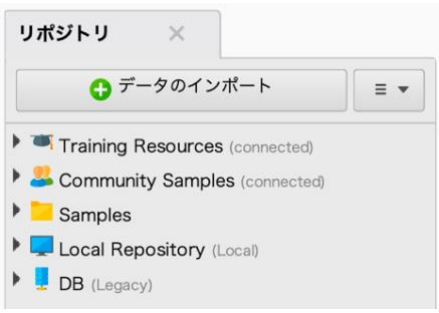
1. RapidMiner Studio を起動します。
2. 画面中央上部の「デザイン（F8）」をクリックして、デザイン画面を開きます。



皆さんはこれから、このデザイン画面での作業に多くの時間を費やすことになります。
デザイン画面で、オペレータを使用し、データマイニングを実行する処理内容を記述します。
（オペレータ：デザイン画面左下の「オペレータ」の中の各アルゴリズム）

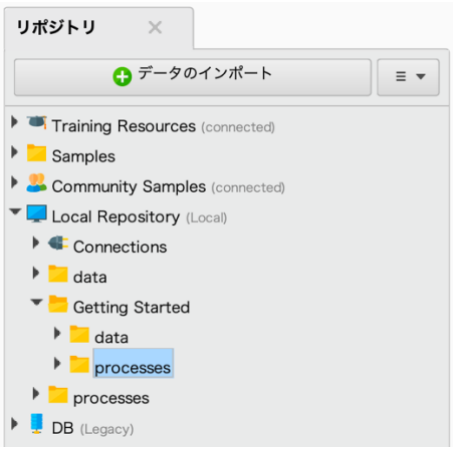
処理を実行した結果は、「結果（F9）」と表示されている結果画面で確認できます。

3. デザイン画面左上の「リポジトリ」は、分析データ、プロセス、そして結果を保管する場所です。



4. 「リポジトリ」に新規フォルダを作ってみましょう。
   a. 「Local Repository」を右クリックします。
   b. 「サブフォルダの作成」を選択します。
   c. 新規フォルダに名前を入力し（例えば「Getting Started」）、「OK」をクリックします。

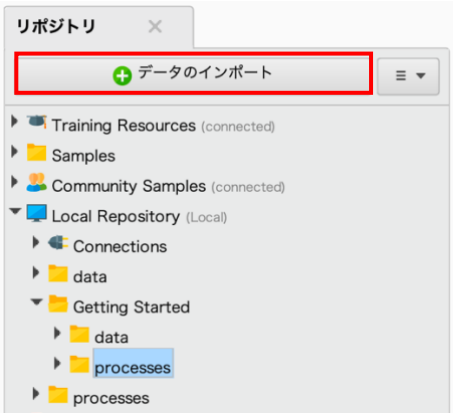
5. 4 の手順を繰り返し（右クリックで新規フォルダを作成、名前入力）、
   「data」フォルダと「processes」フォルダを追加します。
   すると、リポジトリ画面は次のように表示されるでしょう。



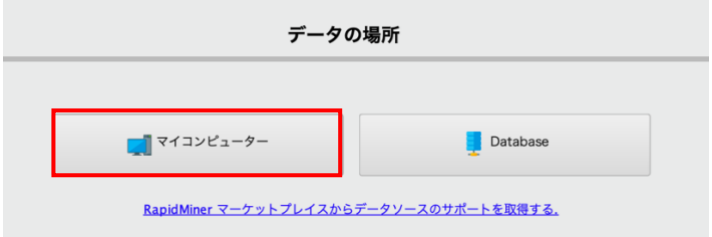
6. データセット「customer-churn-data.xlsx」を、こちらからダウンロードします。

ここでは、データセット「customer-churn-data.xlsx」をリポジトリへ取り込みます。

1. リポジトリ画面左上の[データのインポート」ボタンをクリックします。



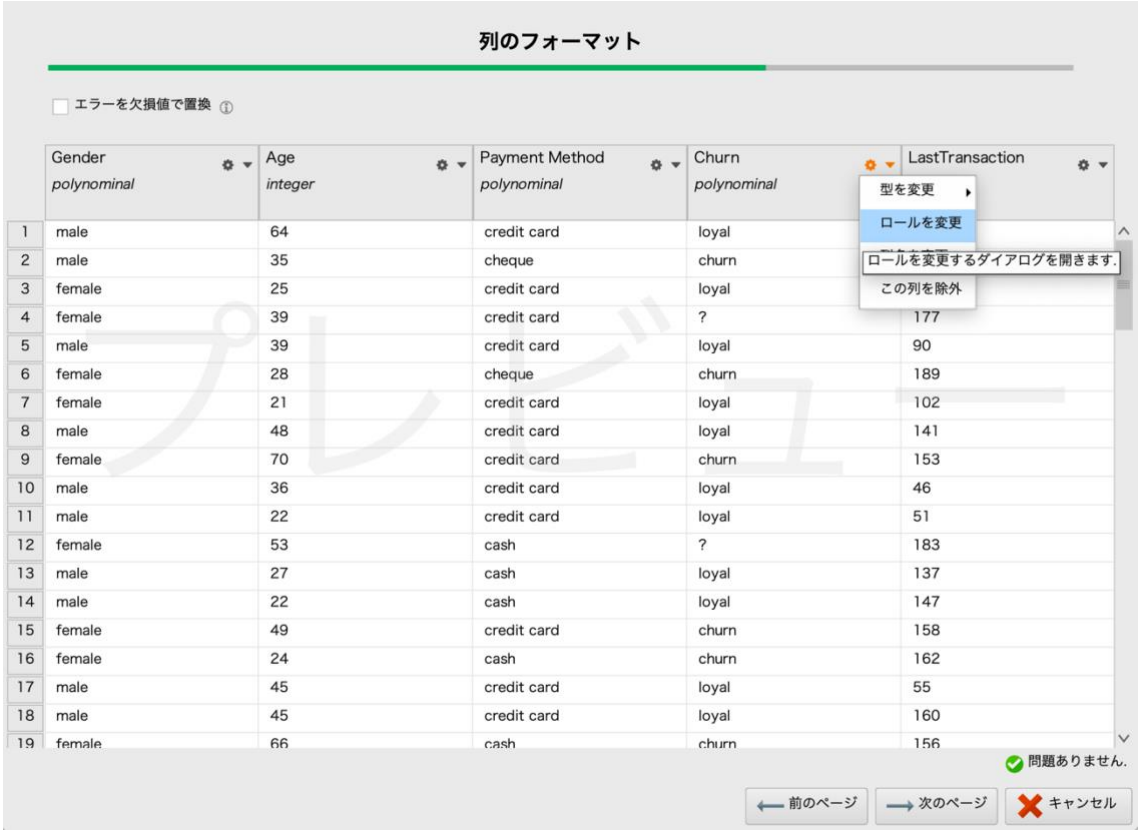
2. 「マイコンピューター」ボタンを選択し、「customer-churn-data.xlsx」を保存した場所（フォルダ）を探し、ファイルを選択し、「次のページ」をクリックします。



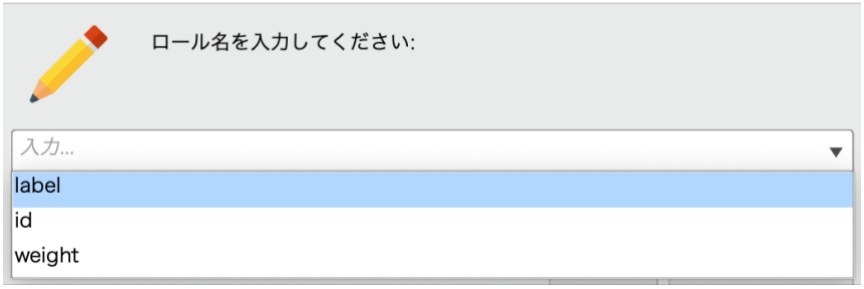
3. リポジトリに取り込むデータを確認します。最初の行は列名を示しています。
今回は標準の設定のまま変更せず、「次のページ」をクリックします。

	A	B	C	D	E
1	Gender	Age	Payment Method	Churn	LastTransaction
2	male	64.000	credit card	loyal	98.000
3	male	35.000	cheque	churn	118.000
4	female	25.000	credit card	loyal	107.000
5	female	39.000	credit card		177.000
6	male	39.000	credit card	loyal	90.000
7	female	28.000	cheque	churn	189.000
8	female	21.000	credit card	loyal	102.000
9	male	48.000	credit card	loyal	141.000
10	female	70.000	credit card	churn	153.000
11	male	36.000	credit card	loyal	46.000
12	male	22.000	credit card	loyal	51.000
13	female	53.000	cash		183.000
14	male	27.000	cash	loyal	137.000
15	male	22.000	cash	loyal	147.000
16	female	49.000	credit card	churn	158.000
17	female	24.000	cash	churn	162.000
18	male	45.000	credit card	loyal	55.000
19	male	45.000	credit card	loyal	160.000
20	female	66.000	cash	churn	156.000
21	female	82.000	cash	churn	177.000

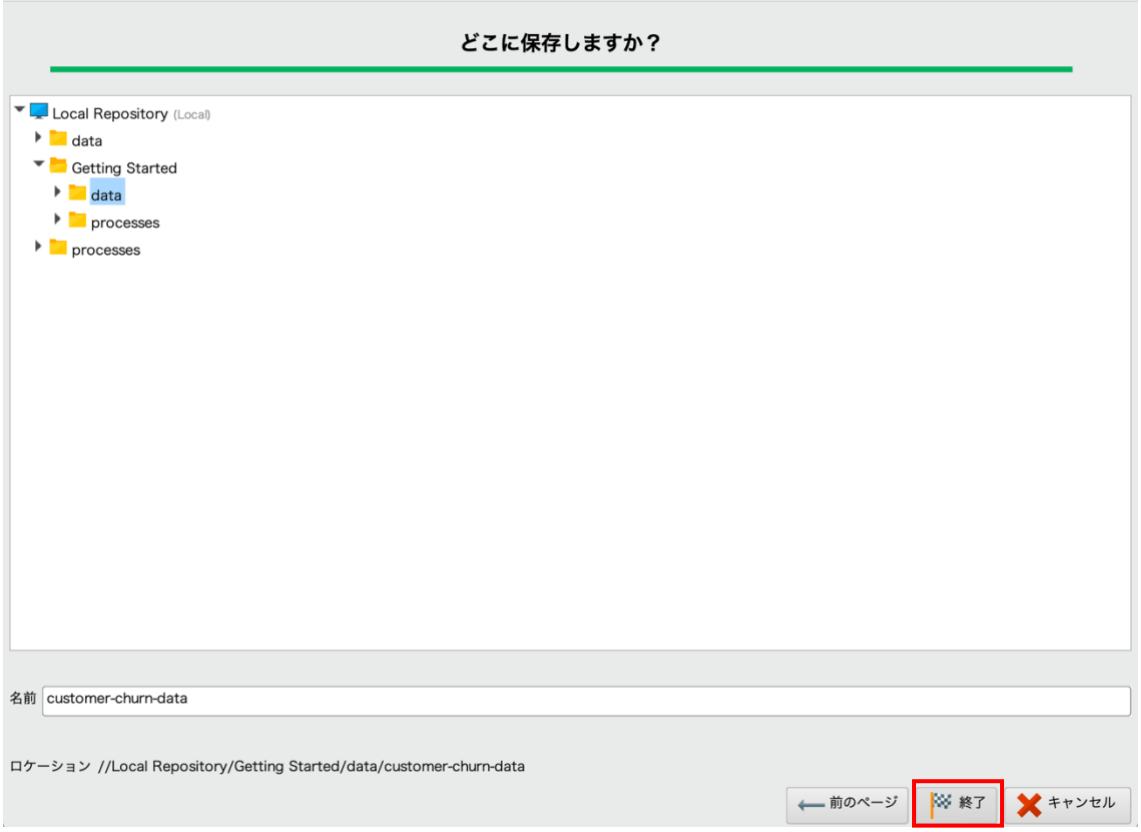
4. 取り込むデータの型や役割を定義します。属性名は、性別(Gender)、年齢(Age)、
支払方法(Payment Method)、離反(Churn)、最終購入日(Last Transaction)です。
「Churn（離反）」は予測する属性ですので、「label（目的変数）」の役割を設定します。
「Churn（離反）」属性の右にある歯車を選択し、「ロールを変更」を選択します。



5. 「label」を選択し、「OK」をクリックします。そして、「次のページ」をクリックします。



6. 「Getting Started」フォルダの中の「data」フォルダを選択し、「終了」をクリックします。
   データはリポジトリに保存されます。



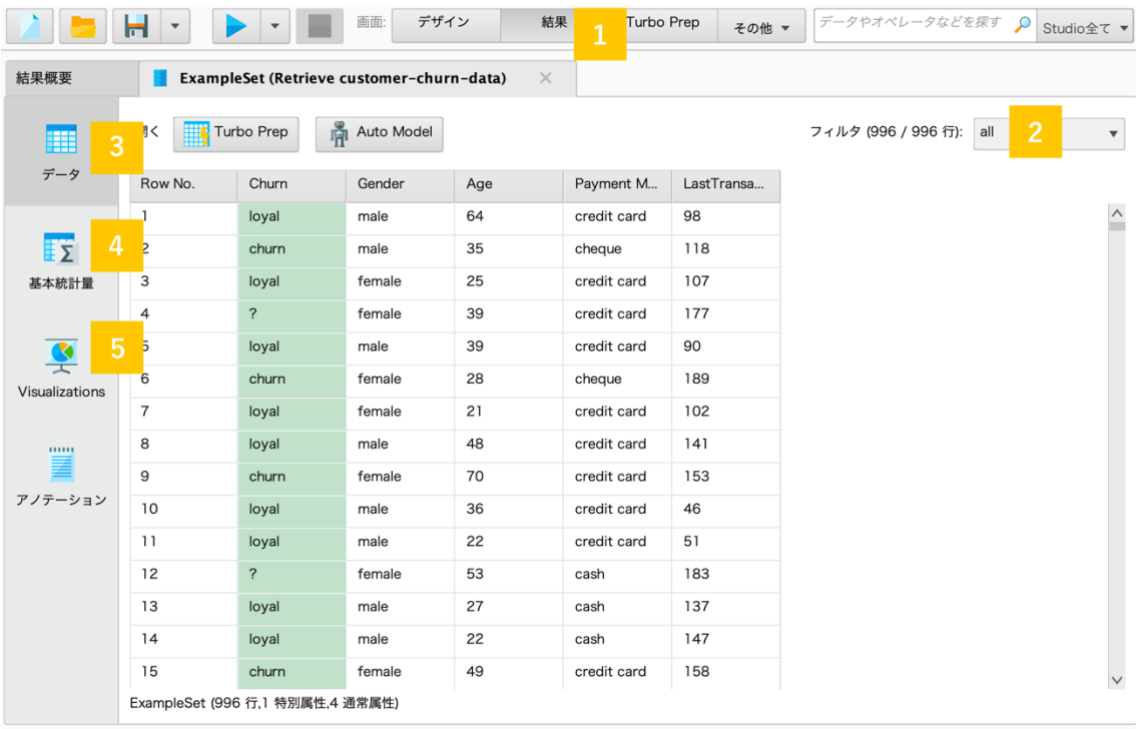
これでデータの取り込みは完了です。

Part2： データ可視化

読み込んだデータを RapidMiner で可視化し、データを俯瞰的に眺め知見を得る方法を学んでいきましょう！RapidMiner を用いて、データと統計、グラフなど各種情報を確認しましょう。

下の図に示す「結果」画面の、5 つの機能を順番に確認していきましょう。

1. 結果
2. フィルタ
3. データ
4. 基本統計量（統計情報）
5. Visualizations（グラフ）



1. 結果

結果画面では、リポジトリに保存している顧客離反データを確認することができます。

結果画面の下部を確認すると、次のことがわかります。

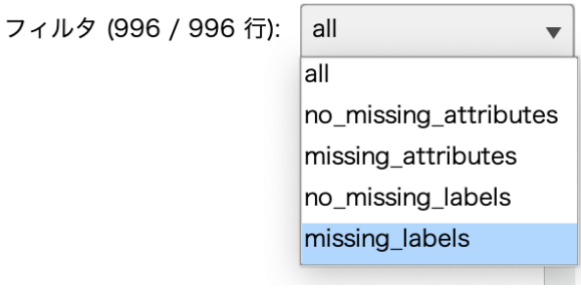
・データセットには、996 行(example)のデータがあります。

・1 つの目的変数(label)と 4 つの説明変数(通常属性)が含まれます。

2. フィルタ

フィルタを使えば、一覧に表示するデータをフィルタリングすることが可能です。

2-1. 下記のように「missing_labels」を選択すると、目的変数(label)の値が「？」表示のレコードのみに絞られます。今回の場合は、996 行のデータのうち 96 行のみになります。



2-2. 他の種類のフィルタも試してみましょう。

3. データ

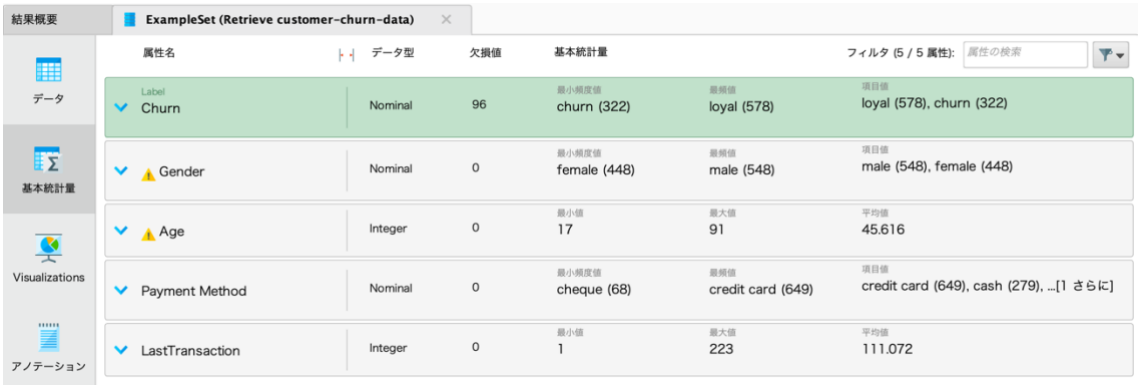
データでは、取り込まれたデータを確認できます。

また、フィルタによってフィルタリングされた結果も反映されます。

4. 基本統計量（統計情報）

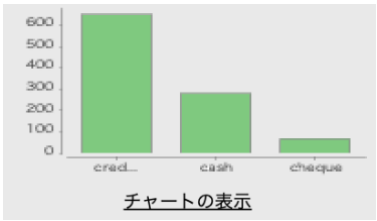
次に、「基本統計量」をクリックします。属性(変数)の型、各属性の欠損値の数、

基本統計量（最小値、最大値、最頻値、平均値、標準偏差など）を確認することができます。



4-1. 「Payment Method（支払方法）」をクリックし、展開してグラフを表示します。

4-2. 「チャートの表示」をクリックします。



チャートの表示

4-3. グラフ画面へ遷移します。「基本統計量」をクリックすると、統計情報の画面へ戻ります。

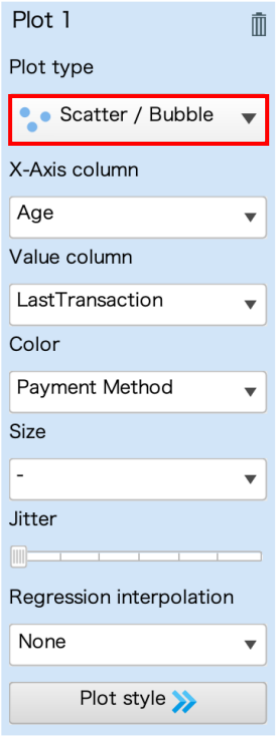
4-4. 「Payment Method（支払方法）」を再度クリックすると元に戻ります。

5. Visualizations（グラフ）

4 で確認したように、基本統計量の「チャートの表示」をクリックすると、

グラフ画面へ遷移します。「基本統計量」下の「Visualizations」をクリックしても遷移できます。

5-1. グラフの種類を選択します。



5-2. 表示されているグラフ名をクリックすると（ここでは「Scatter/Bubble」）、
利用できるグラフがすべて表示されます。新しいグラフでデータを可視化してみましょう。

データを可視化する方法は多くあります。RapidMiner の可視化機能を使って、
時間をかけてデータについて理解しましょう。データの準備が整えば、次はモデルの作成です。

Part3： モデル作成

読み込んだデータから、顧客が離反するか継続するかを予測するモデルを作成する方法を学んでいきましょう！データ間の関連性を見つけ、結果を予測するためのモデルを作成します。後の Part4 で、そのモデルを使用して顧客が離反しそうかを予測します。

モデル作成のために、次のことを行っていきます。

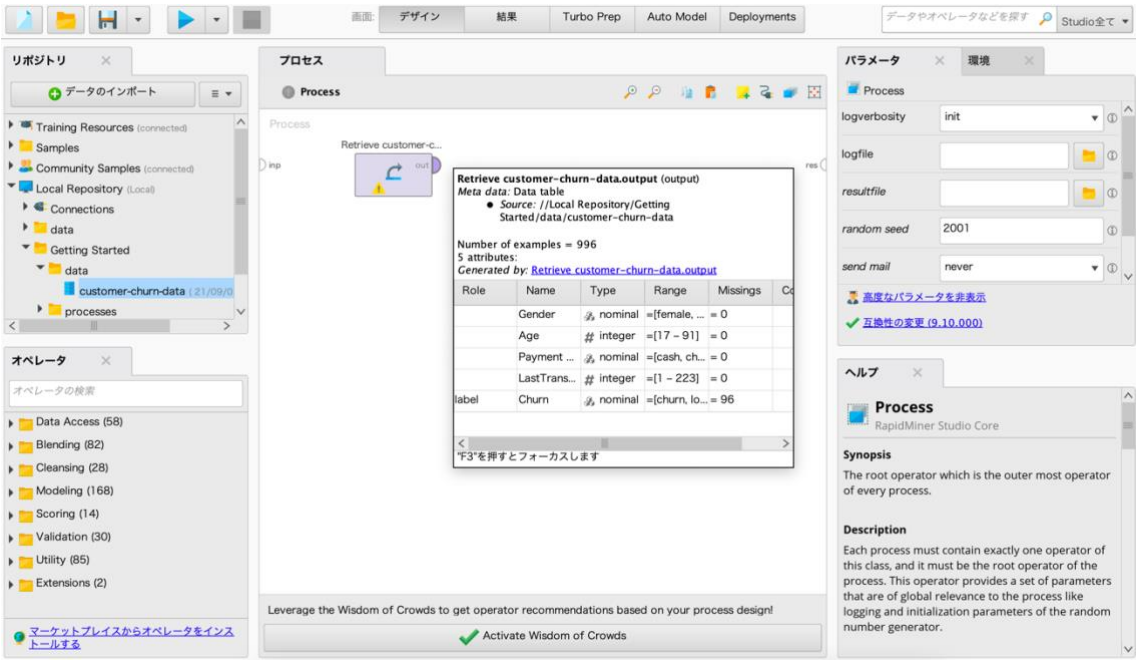
▽ データを読み込みます。
▼ 分析プロセス（以下、プロセス）用に、オペレータ（プロセスの要素）を配置します。
▽ 欠損値データを取り除くために、「Filter Examples」オペレータを追加します。
▼ 「Decision Tree」オペレータを追加します。
▽ プロセスを保存します。

1．データの読み込み

1-1．「結果」画面から「デザイン」画面へ切り替えます。

1-2．リポジトリに取り込んだ「customer-churn-data」をドラッグし、「プロセス」パネルへ配置します。
自動的に「Retrieve」オペレータとして配置されます。

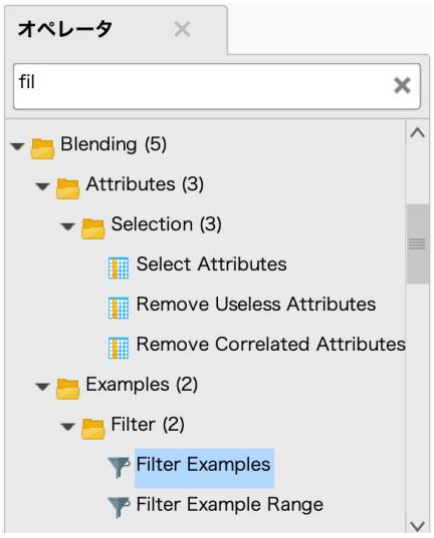
1-3．下図の「Retrieve」オペレータ右側の out(出力)ポート（半円で表示されているもの）に
カーソルを当てると、「customer-churn-data」データの概要を確認することが可能です。

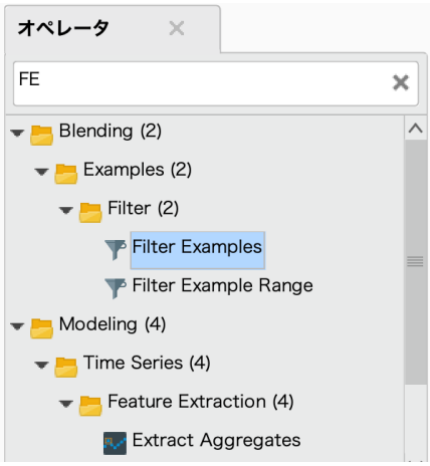


2．欠損値データの除外

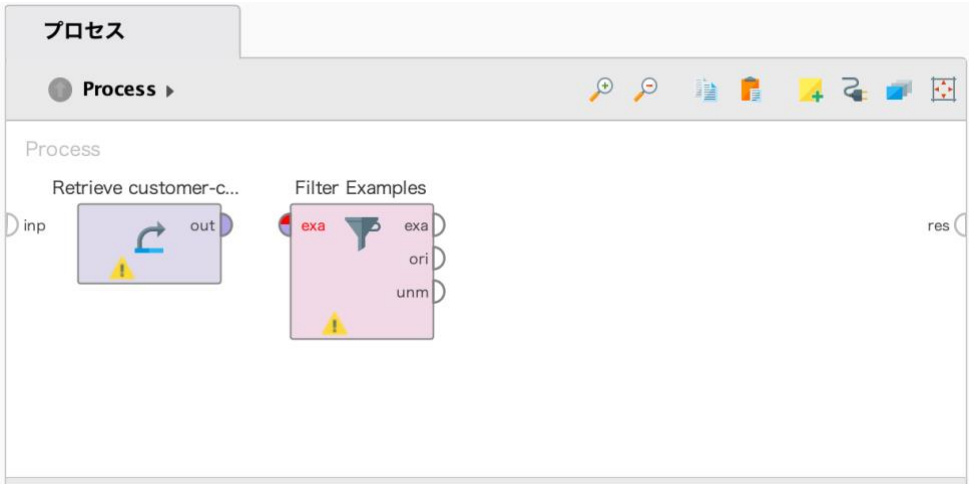
予測モデルを構築するためには、データセットのそれぞれのレコード（今回の場合は顧客単位）に不備がないかチェックする必要があります。言い換えると、予測（モデルの）ルールを作るために、欠損値を持つデータを除外しなければなりません。

2-1. RapidMiner 画面左下「オペレータ」パネルで、検索ボックスにオペレータ名を入力して、「Filter Examples」オペレータを探します。入力文字に関連するオペレータの一覧が表示されます。
右の例の「FE」のように、オペレータ名の省略形を入力しても、一覧に表示されます。





2-2. 「Filter Examples」をドラッグし、「プロセス」パネルへ設置します。



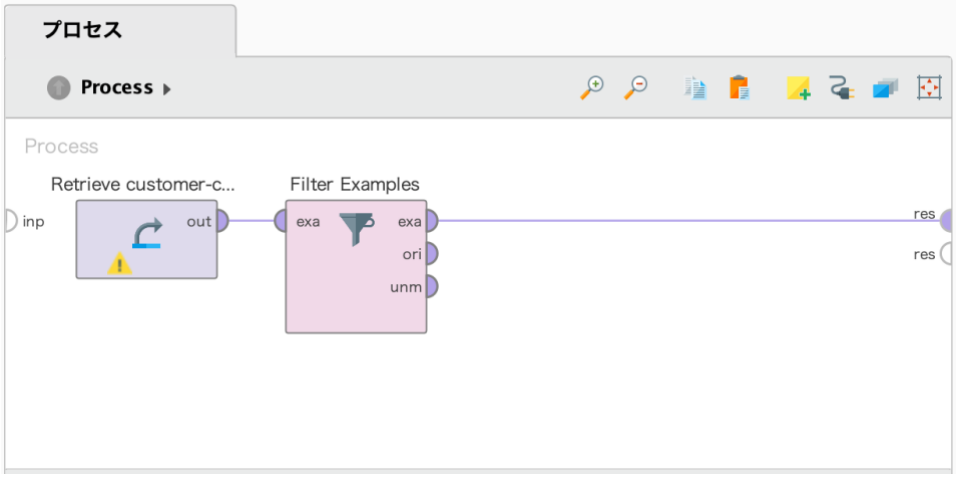
2-3. 「Retrieve」オペレータの出力（out）ポートから、
「Filter Examples」のデータセット（exa）ポートへ線をつなぎます。
出力（out）ポートをクリックして、データセット（exa）ポートへ接続すれば OK です。

2-4. フィルタを設定するために、「Filter Examples」オペレータを選択してクリックします。
すると、RapidMiner 画面右側に、「パラメータ」パネルが表示されます。
「パラメータ」パネルの「フィルタを追加」をクリックします。

2-5. 下図の一番左のプルダウンから、除外したい変数「Churn」を選択します。
2-6. 下図の中間のプルダウンから、「is not missing」を選択します。

2-7. 「OK」ボタンをクリックします。

2-8. 「Filter Examples」オペレータのデータセット（exa）ポートと、
「プロセス」画面右側の結果（res）ポートを接続します。



2-9. RapidMiner 画面上部の「プロセス実行」ボタンをクリックします。

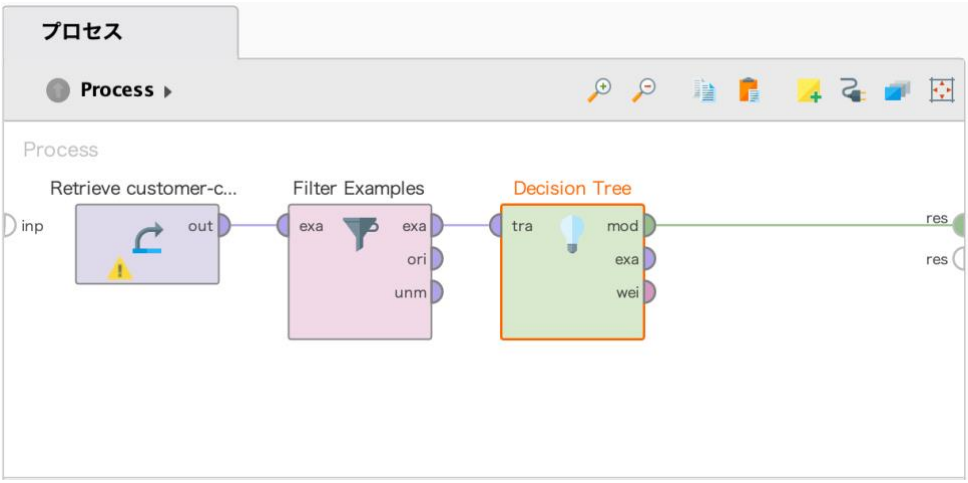
2-10. 結果画面に表示されるレコード数（examples）と、Part2 で表示されたレコード数を比較してみましょう。Part2（欠損値を除外する前）では 996 レコードありましたが、今は 900 レコードになっています。

3. 決定木（Decision Tree）オペレータの追加

今回は決定木（Decision Tree）を使用します。連続した顧客データから、決定木は、説明変数と目的変数間の関係を表わすルールを見つけます。決定木（Decision Tree）オペレータを追加するためには次の作業が必要です。

3-1. 「デザイン」画面に戻ります。

3-2. 「Decision Tree」オペレータを検索し、ドラッグし「Filter Examples」の右へ設置します。
そして、「Filter Examples」右側のデータセット（exa）ポートと、
「Decision Tree」左側の予測モデル生成用データセット（tra）ポートを線で繋ぎます。
さらに、「Decision Tree」右側のモデル（mod）ポートと、結果（res）ポートを線で繋ぎます。



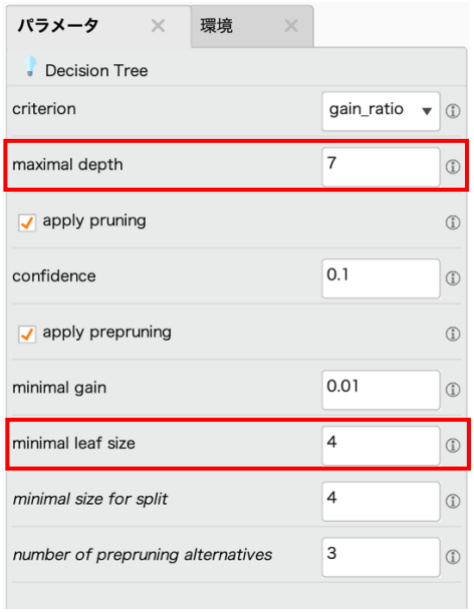
3-3. 全てのパラメータをデフォルト設定のまま、プロセス実行ボタンをクリックします。
※バージョンによって、作成されるモデルは多少異なることがあります。

決定木を生成できました。しかし、これだけ細かく分岐していると、未知のデータ（テストデータ）をモデルに適用させた時にこの通りに分割されるか分かりません。つまり汎化性能が低い可能性があります。
また、解釈性を上げるために、木の深さや葉のサイズの調整を行う「枝刈り」を行う必要があります。

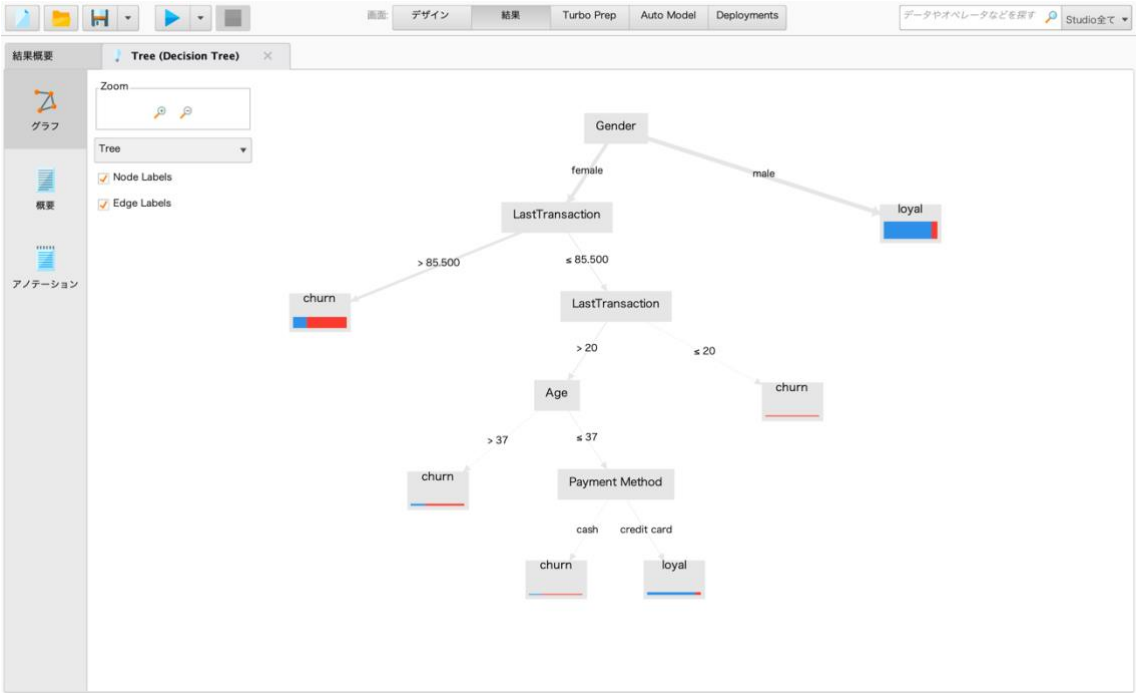
4．枝刈り

木をなるべくシンプルにし、分類をなるべく正確にできるようなパラメータに調整をしましょう。
木の深さは、分割回数のことを表す「maximal depth」で調整ができます。
葉のサイズは、サンプル数を表す「minimal leaf size」で調整ができます。

4-1．結果画面のグラフに表示される、生成した決定木を確認しながらパラメータを変更していきます。
今回、maximal depth=7、minimal leaf size=4 に設定しました。



4-2. パラメータの設定変更ができたらプロセス実行ボタンをクリックします。



5. 木の解釈

最初よりもシンプルな決定木を生成できました。決定木は木の根（root、木の一番上のノード）から始まり、途中各ノードで枝分かれしながら最終の葉（leaf、木の末端のノード）まで伸びています。

5-1. 木の上部から下部を読み解いてみましょう。各「分岐点」の枝ごとに値を持ち、その先の葉では離反(churn)と継続(loyal)の割合が表示されます。

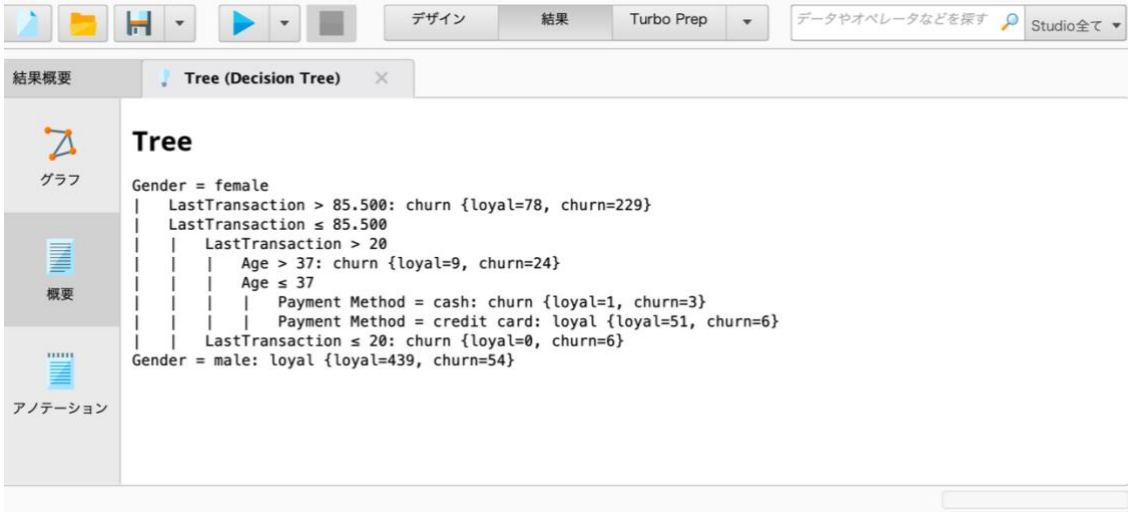
以下は、離反するかどうかを決定木アルゴリズムによりデータから生成したルールです。

・男性は離反しない割合が多い。
・女性かつ最後の購入が 85.5 週より後の場合は離反する割合が多い。
・女性かつ 37 歳より上かつ最後の購入が 20 週から 85.5 週以前の場合は離反する割合が多い。
・女性かつ 37 歳以下かつ最後の購入が 20 週から 85.5 週以前で、
支払方法が現金の場合は離反する割合が多い。
・女性かつ 37 歳以下かつ最後の購入が 20 週から 85.5 週以前でも、
支払方法がクレジットカードの場合は離反しない割合が多い。

5-2. さらに掘り下げると様々なルールを発見できますので、ご自身で確認してみましょう。
全体として、木は 7 つの葉を生成しました。
性別の次に影響力のある変数は最後の購入、その次は年齢、最後は支払い方法の順です。
枝は、1 つの変数から複数本生成されます。

赤と青の比率は、結果を可視化して確認する時に役立ちます。
赤と青は、どれくらいの数の顧客が離反か継続かを示しています。
赤か青のどちらか一方だけで構成されている葉は、より純粋な予測結果と言えます。

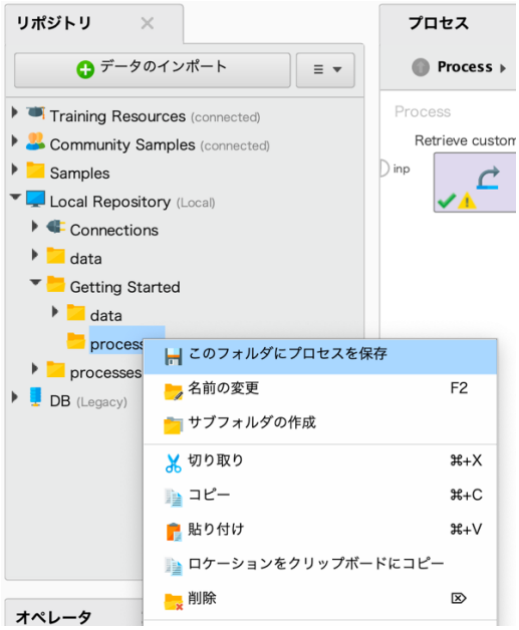
5-3. また、「概要」画面では、結果を数値で確認できます。



6. プロセスの保存
ついに、モデル生成プロセスを保存する時がきました！
6-1. 「デザイン」画面に戻ります。

6-2. 「リポジトリ」画面で、「processes（プロセス）」フォルダを右クリックします。
（Part1 で作成した「Getting Started」フォルダの中です）。

6-3. 「このフォルダにプロセスに保存」をクリックし、結果ダイアログにプロセス名を「Creating training data」のように入力し、「OK」をクリックします。



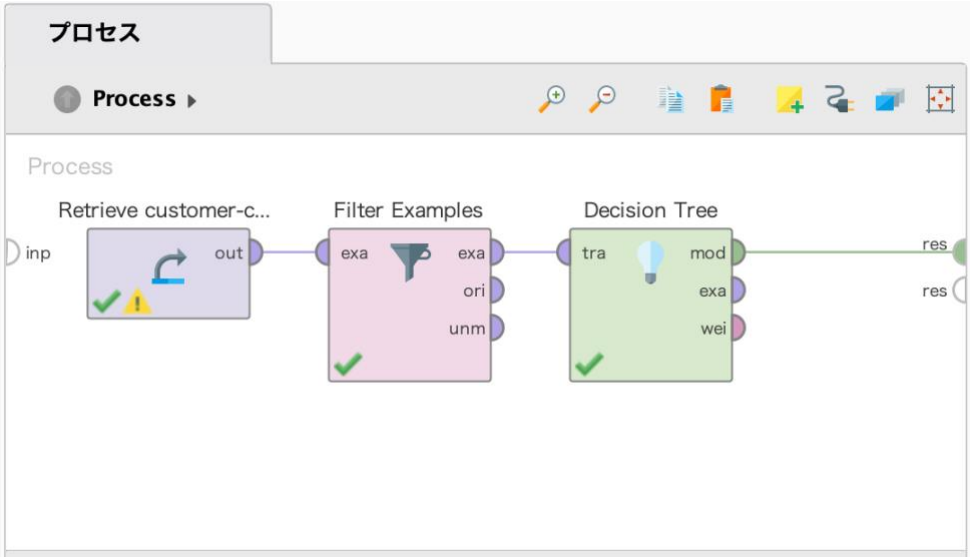
6-4. 「processes（プロセス）」フォルダに、新しいプロセス（歯車のアイコン）が追加されていることを確認してみましょう。

次は、モデルを適用してみましょう！

Part4： モデル適用

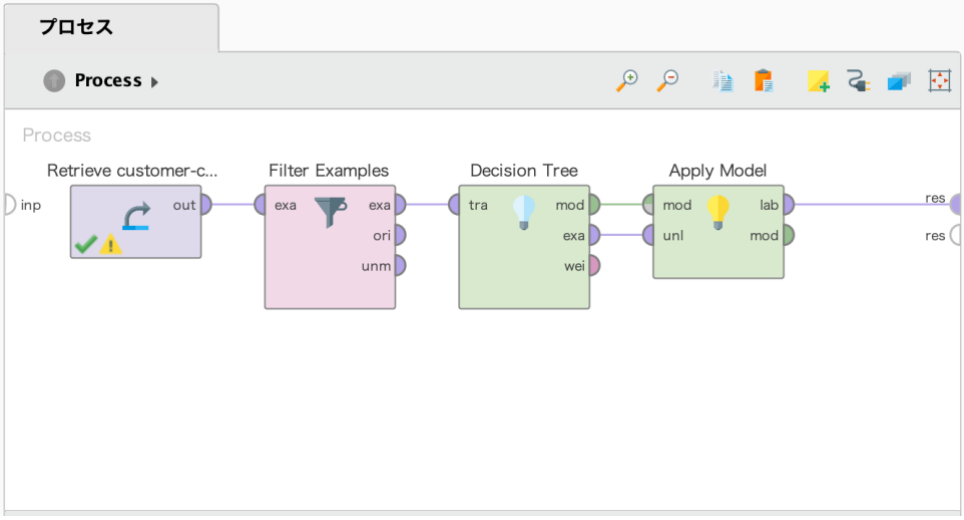
生成したモデルを使って、各顧客が離反するか継続するかを実際に予測してみましょう！
ここでは、Part3 で生成した予測モデルをデータセットに適用させます。

以下の図は Part3 で作成したプロセスです。
顧客データを決定木アルゴリズムに投入し、予測モデルを生成しました。



1．予測モデルの適用

「Apply Model」オペレータを使って、決定木から抽出したルールを、目的変数を持たないデータに適用すると、顧客が離反するかどうかを予測できます。



1-1．「Apply Model」オペレータを検索し、
プロセス内の「Decision Tree」オペレータの右に追加します。

1-2．「Decision Tree」オペレータのモデル（mod）ポートと、「Apply Model」オペレータの（mod）ポートを接続します。

1-3．「Decision Tree」オペレータのモデル（exa）ポートと、「Apply Model」オペレータの目的変数無しの（unl）ポートを接続します。こうすることで、目的変数を持たないデータセットが「Apply Model」オペレータへ渡されます。

1-4. 「Apply Model」オペレータの目的変数有りの（lab）ポートと結果（res）ポートを接続します。
ここまでの操作で、プロセスは上図のように構築できているでしょう。

1-5. 実行ボタンをクリックしてプロセスを実行します。

1-6. 保存ボタン（フロッピー）をクリックし、プロセスを保存します。

2. 結果の理解

プロセスを実行すると、RapidMiner は各データの目的変数（離反するかしないか）を予測します。
結果は次のように表示されます。

Row No.	Churn	prediction(Churn)	confidence(loyal)	confidence(churn)	Gender	Age	Payment Method	LastTransaction
1	loyal	loyal	0.890	0.110	male	64	credit card	98
2	churn	loyal	0.890	0.110	male	35	cheque	118
3	loyal	churn	0.254	0.746	female	25	credit card	107
4	loyal	loyal	0.890	0.110	male	39	credit card	90
5	churn	churn	0.254	0.746	female	28	cheque	189
6	loyal	churn	0.254	0.746	female	21	credit card	102
7	loyal	loyal	0.890	0.110	male	48	credit card	141
8	churn	churn	0.254	0.746	female	70	credit card	153
9	loyal	loyal	0.890	0.110	male	36	credit card	46
10	loyal	loyal	0.890	0.110	male	22	credit card	51
11	loyal	loyal	0.890	0.110	male	27	cash	137
12	loyal	loyal	0.890	0.110	male	22	cash	147
13	churn	churn	0.254	0.746	female	49	credit card	158
14	churn	churn	0.254	0.746	female	24	cash	162
15	loyal	loyal	0.890	0.110	male	45	credit card	55

2-1. 予測モデルの適用により、予測結果と各予測の確信度を表す列が新たに追加されました。
「Churn」項目は目的変数です。

・「prediction(Churn)」は、目的変数の持たないデータセットに対して、顧客が離反するかしないかの予測結果が表示されています。予測結果は決定木ルールをもとに生成されています。

・予測結果の右隣に、各クラスの予測の確信度（confidence）も表示されています。

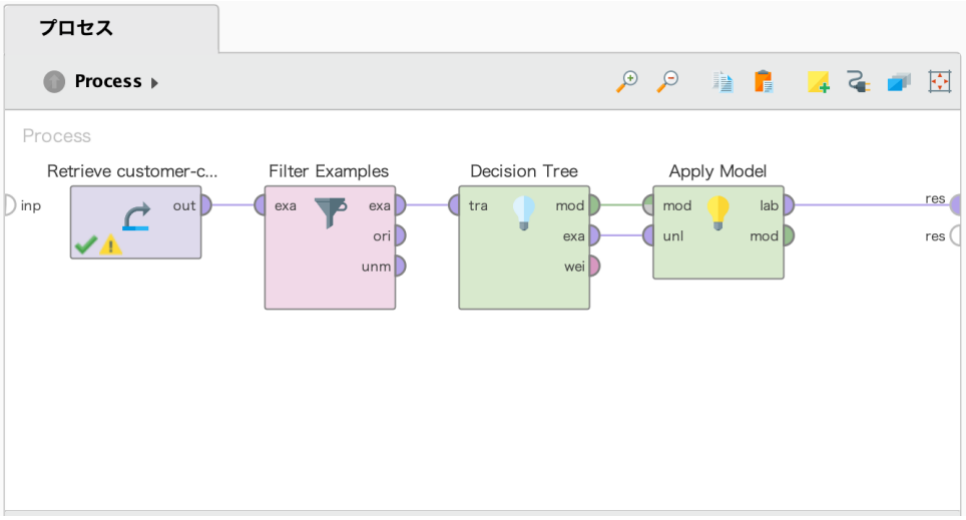
例： 1 行目の confidence(loyal)は、顧客が離反しない可能性が 89.0%であることを表し、confidence(churn)は、顧客が離反する可能性が 11.0%であることを表しています。

次は、結果の精度をより高めるために、モデルの評価を実施します。

Part5： モデル評価

Part4 で作成したモデルが実務に耐えうるものか評価してみましょう！

作成したプロセスを整理すると以下の通りです。
「Decision Tree」オペレータを使って決定木モデルを作成し、
「Apply Model」オペレータを使って各顧客が離反するか継続するか予測しました。



今回は、交差検証（Cross Validation）を使ってモデルを評価します。

1．交差検証

汎化性能（未知のデータに対する性能）を評価する手法です。データ量が十分でない場合でも、全てのデータを有効に活用して複数回検証するので、過小評価も過大評価も含めた精度の平均をとります。どのアルゴリズムを使い、どのパラメータにすれば精度が良いのかを比較を行うことができます。

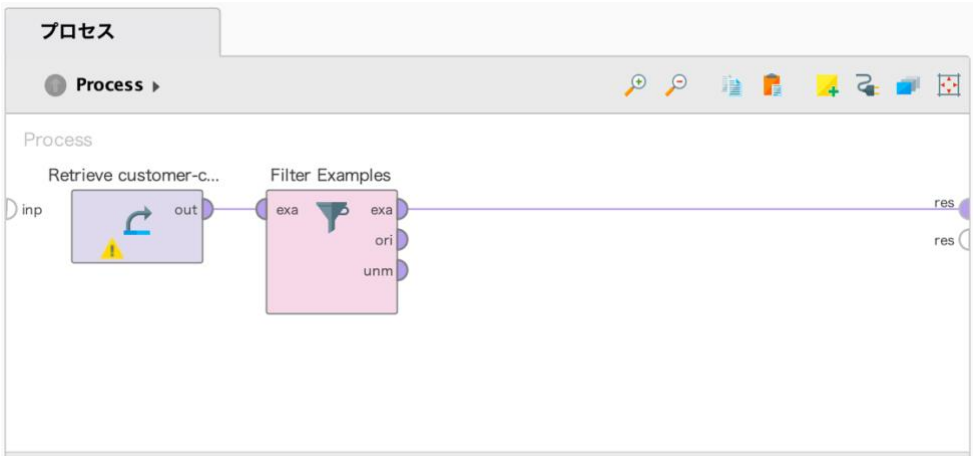
交差検証では、データをいくつかのデータセットに分割します（下図では 5 つの Subset）。

まず、Subset1 をテストデータ（Validation Subset）、
それ以外を訓練用データ（Training Subset）とし、モデルの作成と精度の計算を行います。
次に、Subset2 をテストデータ（Validation Subset）、
それ以外を訓練用データ（Training Subset）とし、モデルの作成と精度の計算を行います。

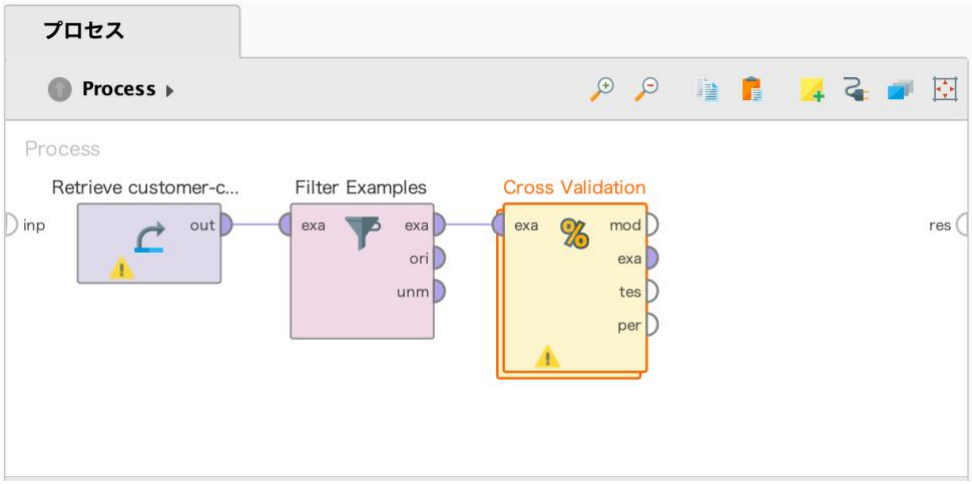
このように、モデルの作成と精度の計算を繰り返し（Iteration）、
全ての精度の平均値を最終的なモデルの精度（Final Accuracy）とします。

早速、RapidMiner で実装してみましょう！

1-1．まず、プロセスを下図（Part3 で作成）の状態にします。不要なオペレータは削除して下さい。



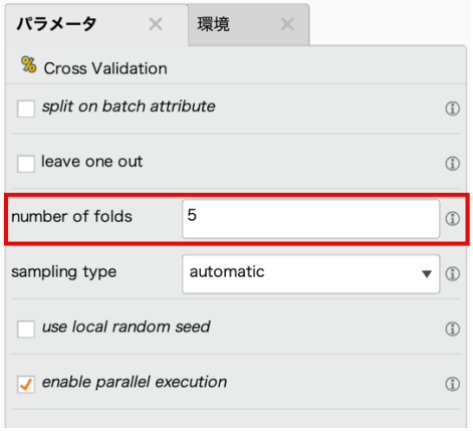
1-2．「Cross Validation」オペレータを配置します。



1-3. データを分割する数（Subset 数）を指定します。

今回、データを 5 分割して交差検証を行うようにパラメータの設定を変更します。

デフォルトでは、10 分割（number of folds=10）の設定になっているので、5 に変更します。



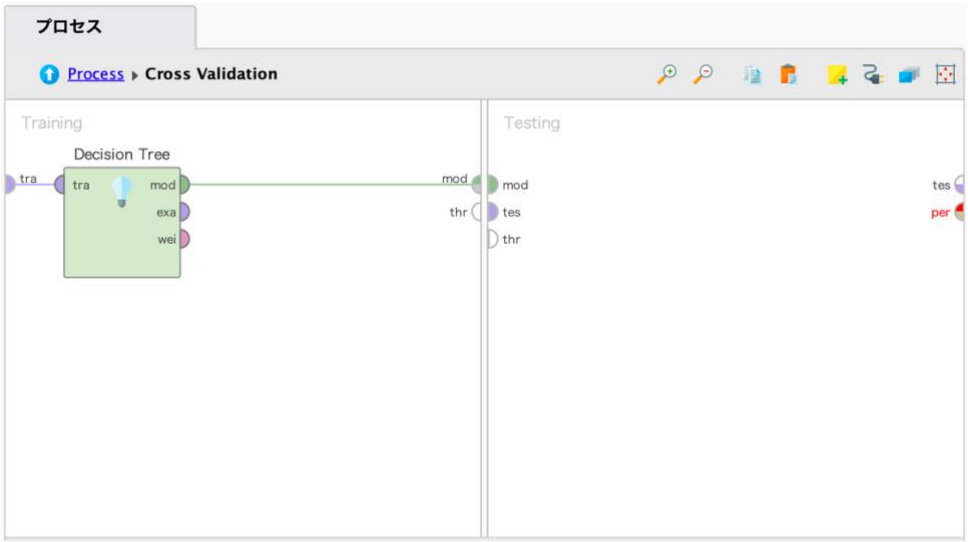
1-4. 配置した「Cross Validation」オペレータをダブルクリックすると、サブプロセス画面が開きます。

「Training」エリア（モデルの作成）と「Testing」エリア（モデルの適用と評価）が表示されます。

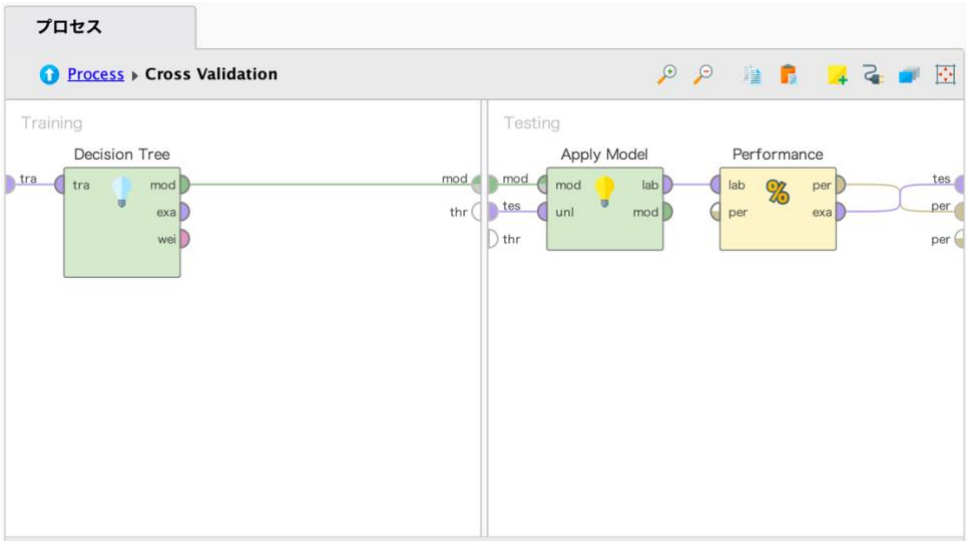
まず、「Training」エリアで、「Decision Tree」オペレータを使って決定木モデルを作成しましょう。

※Decision Tree のパラメータは、Part3、4 と同様に設定します。

（maximal depth=7、minimal leaf size=4）



1-5. 「Testing」エリアで、「Apply Model」オペレータを使って、テストデータに決定木モデルを適用させましょう。続けて、「Performance」オペレータを配置し、線でつなぎます。



1-6. 画面左上の「Process」をクリックし、メインプロセス画面に戻ります。
「Cross Validation」オペレータの「per」ポートと「res」ポートを線で繋いで、
プロセスを実行しましょう。

2. モデルの精度
RapidMiner では「Performance」オペレータを使ってモデルの精度を計算できます。
先ほど実行した結果を確認してみましょう。

2-1. モデルの精度は 83.33%、標準偏差は 1.76%です。
精度の良い（当てはまりの良い）モデルであることが分かります。

accuracy: 83.33% +/- 1.76% (micro average: 83.33%)

	true loyal	true churn	class precision
pred. loyal	501	73	87.28%
pred. churn	77	249	76.38%
class recall	86.68%	77.33%	

※交差検証の実行結果は、テストデータの選び方によって異なる場合があります。

2-2. まず左の枠内の再現率（class recall）を確認してみましょう。

「離反しない」（true loyal）に対し「離反しない」と予測した（pred.loyal）顧客は 501 人、「離反しない」（true loyal）に対し「離反する」と予測した（pred.churn）顧客は 77 人です。よって再現率は、501／（501＋77）＝0.8668 となり、86.68%であることが分かります。

同様に、右の枠内の再現率も計算できます。

accuracy: 83.33% +/- 1.76% (micro average: 83.33%)

	true loyal	true churn	class precision
pred. loyal	501	73	87.28%
pred. churn	77	249	76.38%
class recall	86.68%	77.33%	

2-3. 下段の枠内の適合率（class precision）を確認してみましょう。

「離反しない」（true loyal）に対し「離反する」と予測した（pred.churn）顧客は 77 人、「離反する」（true churn）に対し「離反する」と予測した（pred.churn）顧客は 249 人です。よって適合率は、249／（77＋249）＝0.7638 となり、76.38%であることが分かります。

accuracy: 83.33% +/- 1.76% (micro average: 83.33%)

	true loyal	true churn	class precision
pred. loyal	501	73	87.28%
pred. churn	77	249	76.38%
class recall	86.68%	77.33%	

さいごに

RapidMiner で分析データの読込、可視化、モデル作成・評価する方法を一通りご紹介しました。
皆さまがお持ちのデータを使ってぜひ予測分析を試してみてください。