

Extensions IBM SPSS Modeler

IBM

Remarque

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations figurant à la section «Remarques», à la page 61.

Informations sur le produit

LE PRESENT DOCUMENT EST LIVRE EN L'ETAT SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFAÇON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE.

Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. Les informations qui y sont fournies sont susceptibles d'être modifiées avant que les produits décrits ne deviennent eux-mêmes disponibles. En outre, il peut contenir des informations ou des références concernant certains produits, logiciels ou services non annoncés dans ce pays. Cela ne signifie cependant pas qu'ils y seront annoncés.

Pour plus de détails, pour toute demande d'ordre technique, ou pour obtenir des exemplaires de documents IBM, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial.

Vous pouvez également consulter les serveurs Internet suivants :

- <http://www.fr.ibm.com> (serveur IBM en France)
- <http://www.ibm.com/ca/fr> (serveur IBM au Canada)
- <http://www.ibm.com> (serveur IBM aux Etats-Unis)

*Compagnie IBM France
Direction Qualité
17, avenue de l'Europe
92275 Bois-Colombes Cedex*

© Copyright IBM France 2017. Tous droits réservés.

Cette édition s'applique à la version 18.0.0 d'IBM SPSS Modeler et à toutes les éditions et modifications ultérieures sauf mention contraire dans les nouvelles éditions.

Table des matières

Avis aux lecteurs canadiens	v
Chapitre 1. Langages pris en charge	1
R	1
Python for Spark	1
Scriptage avec Python for Spark	2
Noeuds d'extension.	12
Noeud Exportation d'extension.	12
Noeud Sortie d'extension.	14
noeud Modèle d'extension	16
Nugget de modèle Extension	17
Noeud de transformation d'extension.	20
Noeud Importation d'extension.	21
Chapitre 2. Extensions	23
Hub d'extension.	23
Onglet Explorer	24
Onglet Installé	25
Onglet Paramètres	25
Détails de l'extension	26
Installation des bundles d'extension locaux.	26
Emplacements de l'installation des extensions	26
Packs R nécessaires.	27
Création et gestion des noeuds personnalisés	27
Présentation du générateur de boîtes de dialogue personnalisées	28
Création d'une boîte de dialogue de noeud personnalisée.	29
Propriétés des boîtes de dialogue	29
Présentation des contrôles dans le canevas de boîte de dialogue	30
Création du modèle de script	31
Prévisualisation d'une boîte de dialogue de noeud personnalisée	32
Types de contrôle	32
Propriétés d'extension	52
Gestion des boîtes de dialogue de noeud personnalisées	54
Création de versions localisées des boîtes de dialogue de noeud personnalisées	57
Importation et exportation de données à l'aide de Python for Spark	59
Importation et exportation de données à l'aide de R.	59
Remarques	61
Marques	62
Dispositions relatives à la documentation du produit	63
Index	65

Avis aux lecteurs canadiens

Le présent document a été traduit en France. Voici les principales différences et particularités dont vous devez tenir compte.

Illustrations

Les illustrations sont fournies à titre d'exemple. Certaines peuvent contenir des données propres à la France.

Terminologie

La terminologie des titres IBM peut différer d'un pays à l'autre. Reportez-vous au tableau ci-dessous, au besoin.

IBM France	IBM Canada
ingénieur commercial	représentant
agence commerciale	succursale
ingénieur technico-commercial	informaticien
inspecteur	technicien du matériel

Claviers

Les lettres sont disposées différemment : le clavier français est de type AZERTY, et le clavier français-canadien de type QWERTY.








OS/2 et Windows - Paramètres canadiens

Au Canada, on utilise :

- les pages de codes 850 (multilingue) et 863 (français-canadien),
- le code pays 002,
- le code clavier CF.

Nomenclature

Les touches présentées dans le tableau d'équivalence suivant sont libellées différemment selon qu'il s'agit du clavier de la France, du clavier du Canada ou du clavier des États-Unis. Reportez-vous à ce tableau pour faire correspondre les touches françaises figurant dans le présent document aux touches de votre clavier.

France	Canada	Etats-Unis
 (Pos1)		Home
Fin	Fin	End
 (PgAr)		PgUp
 (PgAv)		PgDn
Inser	Inser	Ins
Suppr	Suppr	Del
Echap	Echap	Esc
Attn	Intrp	Break
Impr écran	ImpEc	PrtSc
Verr num	Num	Num Lock
Arrêt défil	Défil	Scroll Lock
 (Verr maj)	FixMaj	Caps Lock
AltGr	AltCar	Alt (à droite)

Brevets

Il est possible qu'IBM détienne des brevets ou qu'elle ait déposé des demandes de brevets portant sur certains sujets abordés dans ce document. Le fait qu'IBM vous fournisse le présent document ne signifie pas qu'elle vous accorde un permis d'utilisation de ces brevets. Vous pouvez envoyer, par écrit, vos demandes de renseignements relatives aux permis d'utilisation au directeur général des relations commerciales d'IBM, 3600 Steeles Avenue East, Markham, Ontario, L3R 9Z7.

Assistance téléphonique

Si vous avez besoin d'assistance ou si vous voulez commander du matériel, des logiciels et des publications IBM, contactez IBM direct au 1 800 465-1234.

Chapitre 1. Langages pris en charge

IBM® SPSS Modeler prend en charge R et Apache Spark (via Python). Consultez les sections suivantes pour plus d'informations.

R

IBM SPSS Modeler prend en charge R.

Syntaxe admise

- Dans le champ de syntaxe de l'onglet **Syntaxe** des différents noeuds Extension, seules les déclarations et fonctions reconnues par R sont admises.
- Pour le noeud Transformation Extension et le nugget de modèle d'extension, les données sont envoyées via le script R (par lots). C'est la raison pour laquelle, les scripts R pour le modèle de scoring et les noeuds de processus ne doivent pas contenir d'opérations couvrant ou combinant des lignes dans les données, comme le tri ou l'agrégation. Cette limitation est imposée pour faire en sorte que les données puissent être scindées dans un environnement Hadoop et au cours de l'exploration de base de données. Les noeuds Sortie d'extension et les noeuds de génération de modèle d'extension ne sont pas concernés par cette limitation.
- L'ajout d'un mode de transfert de données autre que par lots dans le noeud Transformation Extension et dans le nugget de modèle Extension signifie que vous pouvez déployer ou combiner les lignes des données de SPSS Modeler Server.
- Tous les noeuds R sont des environnements R globaux indépendants. Par conséquent, l'utilisation des fonctions `library` dans deux noeuds R distincts nécessite de charger la bibliothèque R dans les deux scripts R.
- Pour afficher la valeur d'un objet R défini dans le script R, vous devez inclure un appel à la fonction `print`. Par exemple, pour afficher la valeur de l'objet R `data` (données), incluez la ligne suivante dans le script R :

```
print(data)
```
- Vous ne pouvez pas inclure un appel à la fonction R `setwd` dans le script R, car cette fonction est utilisée par IBM SPSS Modeler pour contrôler le chemin d'accès au fichier de sortie des scripts R.
- Les paramètres de flux définis pour être utilisés dans les expressions et le scriptage CLEM ne sont pas reconnus s'ils sont inclus dans les scripts R.

Python for Spark

IBM SPSS Modeler prend en charge les scripts Python pour Apache Spark.

Remarque :

- Les noeuds Python dépendent de l'environnement Spark.
- Les scripts Python doivent utiliser l'API Spark car les données sont présentées sous la forme d'un Spark DataFrame.
- Les anciens noeuds créés dans la version 17.1 ne seront toujours exécutés que sur IBM SPSS Analytic Server (les données proviennent d'un noeud source IBM SPSS Analytic Server et n'ont pas été extraites dans IBM SPSS Modeler Server). Les nouveaux noeuds Python et Générateur de boîtes de dialogue personnalisées créés dans la version 18.0 ou une version ultérieure peuvent être exécutés sur IBM SPSS Modeler Server.
- Lors de l'installation de Python, assurez-vous que tous les utilisateurs disposent de droits d'accès à l'installation de Python.

- Si vous souhaitez utiliser Machine Learning Library (MLlib), vous devez installer une version de Python qui inclut NumPy. Vous devez ensuite configurer IBM SPSS Modeler Server (ou le serveur local dans IBM SPSS Modeler Client) pour utiliser votre installation Python. Pour plus d'informations, voir «Scriptage avec Python for Spark».

Scriptage avec Python for Spark

IBM SPSS Modeler peut exécuter des scripts Python en utilisant l'infrastructure Apache Spark pour traiter les données. Cette documentation fournit la description de l'API Python pour les interfaces fournies.

L'installation IBM SPSS Modeler inclut une distribution Spark (par exemple, IBM SPSS Modeler 18.1 inclut Spark 2.1.0). Si vous préférez utiliser une version différente de Spark, vous pouvez la configurer en ajoutant les paramètres `eas_spark_path` et `eas_spark_version` à `options.cfg`. Les valeurs possibles pour le paramètre `eas_spark_version` sont 1.x ou 2.x. Exemple :

```
eas_spark_path, "C:/spark_1.5"
eas_spark_version, "1.x"
```

Conditions requises

- Si vous prévoyez d'exécuter des scripts Python/Spark sur IBM SPSS Analytic Server, vous devez disposer d'une connexion à Analytic Server et Analytic Server doit avoir accès à une installation compatible d'Apache Spark. Pour plus d'informations sur l'utilisation d'Apache Spark comme moteur d'exécution, voir la documentation IBM SPSS Analytic Server.
- Si vous prévoyez d'exécuter des scripts Python/Spark sur IBM SPSS Modeler Server (ou sur le serveur local inclus avec IBM SPSS Modeler Client, qui nécessite Windows 64, Linux64 ou Mac64), vous n'avez plus besoin d'installer Python et d'éditer le fichier `options.cfg` pour utiliser votre installation Python. A compter de la version 18.1, IBM SPSS Modeler inclut une distribution Python. Cependant, si vous avez besoin d'un module non inclus avec la distribution Python IBM SPSS Modeler par défaut, vous avez la possibilité d'accéder au <répertoire_installation_Modeler>/python et d'installer des packages supplémentaires.

Même si une distribution Python est désormais incluse avec IBM SPSS Modeler, vous pouvez toujours pointer sur votre propre installation Python comme dans les éditions précédentes, en ajoutant l'option suivante dans le fichier `options.cfg` :

```
# Définie sur le chemin d'accès complet de l'exécutable de python
(avec le nom de l'exécutable) pour permettre l'utilisation de PySpark.
eas_pyspark_python_path, ""
```

Par exemple :

```
eas_pyspark_python_path, "C:/Your_Python_Install/python.exe"
```

- Si vous utilisez le client IBM SPSS Modeler sous Linux, créez un dossier pour stocker les extensions du générateur de boîtes de dialogue personnalisées. Ajoutez ensuite la commande suivante au fichier `modelerclient.sh`, <votre_dossier_CDB> correspondant au dossier que vous avez créé :
- ```
export IBM_SPSS_MODELER_EXTENSION_PATH=<votre_dossier_CDB>
```

## Objet contextuel IBM SPSS Analytic Server

Le contexte d'exécution d'un script Python/Spark est défini par un objet contextuel Analytic Server. Lors d'une exécution sur IBM SPSS Modeler Server, l'objet contextuel est destiné à la version imbriquée d'Analytic Server qui est incluse avec l'installation d'IBM SPSS Modeler Server. Pour obtenir l'objet contextuel, le script doit inclure ce qui suit :

```
import spss.pyspark.runtime
asContext = spss.pyspark.runtime.getContext()
```

A partir du contexte Analytic Server, vous pouvez obtenir le contexte Spark et le contexte SQL :

```
sparkContext = asc.getSparkContext()
sqlContext = asc.getSparkSQLContext()
```



Pour plus d'informations sur le contexte Spark et le contexte SQL, reportez-vous à la documentation Apache Spark.

## Accès aux données

Les données sont transférées entre un script Python/Spark et le contexte d'exécution sous la forme d'une trame de données Spark SQL DataFrame. Un script qui utilise des données (c'est-à-dire tout noeud à l'exception d'un noeud source) doit extraire la trame de données à partir du contexte :

```
inputData = asContext.getSparkInputData()
```

Un script qui génère des données (c'est-à-dire tout noeud à l'exception d'un noeud terminal) doit renvoyer une trame de données au contexte :

```
asContext.setSparkOutputData(outputData)
```

Vous pouvez utiliser le contexte SQL pour créer une trame de données de sortie à partir d'un RDD lorsque cela est nécessaire :

```
outputData = sqlContext.createDataFrame(rdd)
```

## Définition du modèle de données

Un noeud qui génère des données doit également définir un modèle de données décrivant les champs du noeud visibles en aval. Dans la terminologie Spark SQL, le modèle de données est le schéma.

Un script Python/Spark définit son modèle de données de sortie sous la forme d'un objet `pyspark.sql.types.StructType`. Un élément `StructType` décrit une ligne de la trame de données de sortie et est construit à partir d'une liste d'objets `StructField`. Chaque `StructField` décrit un champ unique du modèle de données de sortie.

Vous pouvez obtenir le modèle de données des données d'entrée à l'aide de l'attribut `:schema` de la trame de données d'entrée :

```
inputSchema = inputData.schema
```

Les champs qui sont transmis inchangés peuvent être copiés du modèle de données d'entrée dans le modèle de données de sortie. Les champs qui sont nouveaux ou modifiés dans le modèle de données de sortie peuvent être créés à l'aide du constructeur `StructField` :

```
field = StructField(name, dataType, nullable=True, metadata=None)
```

Pour plus d'informations sur le constructeur, reportez-vous à la documentation Spark.

Vous devez indiquer au moins le nom de champ et son type de données. Vous pouvez aussi spécifier des métadonnées pour fournir une mesure, un rôle et une description pour le champ (voir «Métadonnées de données», à la page 7).

## Mode DataModelOnly

IBM SPSS Modeler doit connaître le modèle de données de sortie d'un noeud, avant l'exécution de ce dernier, pour pouvoir activer l'édition en aval. Pour obtenir le modèle de données de sortie d'un noeud Python/Spark, IBM SPSS Modeler exécute le script dans un mode "data model only" spécial dans lequel aucune donnée n'est disponible. Le script peut identifier ce noeud à l'aide de la méthode `isComputeDataModelOnly` sur l'objet contextuel `Analytic Server`.

Le script d'un noeud de transformation peut suivre le modèle général suivant :

```
if asContext.isComputeDataModelOnly():
 inputSchema = asContext.getSparkInputSchema()
 outputSchema = ... # construct the output data model
```

```

asContext.setSparkOutputSchema(outputSchema)
else:
 inputData = asContext.getSparkInputData()
 outputData = ... # construct the output data frame
 asContext.setSparkOutputData(outputData)

```

## Création d'un modèle

Un noeud qui crée un modèle doit renvoyer au contexte d'exécution un élément content décrivant suffisamment le modèle pour que le noeud qui applique le modèle puisse le recréer à l'identique ultérieurement.

Le contenu du modèle est défini en terme de paires clé/valeur pour lesquels la signification des clés et des valeurs est uniquement connue des noeuds de construction et de scoring et n'est pas interprétée par Modeler. Le noeud peut éventuellement affecter un type MIME à la valeur afin que Modeler puisse afficher à l'utilisateur ces valeurs (dont les types sont connus) dans le nugget de modèle.

Une valeur dans ce contexte peut être une valeur PMML, HTML, une image, etc. Pour ajouter un valeur au contenu de modèle (dans le script de génération) :

```
asContext.setModelContentFromString(key, value, mimeType=None)
```

Pour extraire une valeur du contenu de modèle (dans le script de score) :

```
value = asContext.getModelContentToString(key)
```

En raccourci, lorsqu'un modèle ou une partie de modèle est stocké dans un fichier ou un dossier du système de fichiers, vous pouvez regrouper tous les contenus stockés dans cet emplacement en un seul appel (dans le script de génération) :

```
asContext.setModelContentFromPath(key, path)
```

Notez que dans ce cas, il n'est pas possible de spécifier un type MIME car le regroupement peut contenir plusieurs types de contenus.

Si vous avez besoin d'un emplacement temporaire pour stocker le contenu pendant la création du modèle, vous pouvez obtenir un emplacement approprié dans le contexte :

```
path = asContext.createTemporaryFolder()
```

Pour extraire le contenu existant vers un emplacement temporaire du système de fichiers (dans le script de score) :

```
path = asContext.getModelContentToPath(key)
```

## Traitement des erreurs

Pour déclencher des erreurs, lancez une exception à partir du script et affichez-la pour l'utilisateur IBM SPSS Modeler. Certaines exceptions sont prédéfinies dans le module `spss.pyspark.exceptions`. Par exemple :

```

from spss.pyspark.exceptions import ASContextException
if ... some error condition ...:
 raise ASContextException("message to display to user")

```

## Contexte Analytic Server

Le contexte prend en charge l'interface contextuelle Analytic Server pour l'interaction avec IBM SPSS Analytic Server.

## Objets `AnalyticServerContext`

Les objets `AnalyticServerContext` définissent l'environnement contextuel qui fournit plusieurs interfaces pour l'interaction avec IBM SPSS Analytic Server. Une application qui veut générer cette instance de contexte doit le faire en utilisant l'interface `spss.pyspark.runtime.getContext()` et non en implémentant directement l'interface.

Renvoie l'instance Pyspark python `SparkContext` :

```
cxt.getSparkContext() : SparkContext
```

Renvoie l'instance Pyspark python `SQLContext` :

```
cxt.getSparkSQLContext() : SQLContext
```

Renvoie `True` pour indiquer que l'exécution a uniquement pour but de calculer le modèle de données de sortie. Sinon, renvoie `False` :

```
cxt.isComputeDataModelOnly() : Boolean
```

Renvoie `True` si le script s'exécute dans l'environnement Spark. Actuellement, il renvoie toujours `True` :

```
cxt.isSparkExecution() : Boolean
```

Charge les données d'entrée à partir du fichier temporaire en amont et générer l'instance `pyspark.sql.DataFrame` :

```
cxt.getSparkInputData() : DataFrame
```

Renvoie une instance `pyspark.sql.StructType` générée à partir du modèle de données d'entrée. Renvoie `None` si le modèle de données d'entrée n'existe pas :

```
cxt.getSparkInputSchema() : StructType
```

Sérialise la trame de données de sortie dans le contexte Analytic Server et renvoie le contexte :

```
cxt.setSparkOutputData(outDF) : AnalyticServerContext
```

Paramètre :

- `outDF (DataFrame)` : valeur de la trame de données de sortie

Exceptions :

- `DataOutputNotSupported` : si cette interface est appelée dans la fonction `pyspark:buildmodel`
- `ASContextException` : si la trame de données de sortie est `None`
- `InconsistentOutputDataModel` : les noms de champs et le type de stockage communs aux deux objets sont incohérents

Convertit l'instance `outSchema StructType` en un modèle de données, le sérialise dans le contexte Analytic Server et renvoie le contexte :

```
cxt.setSparkOutputSchema(outSchema) : AnalyticServerContext
```

Paramètre :

- `outSchema(StructType)` : objet `StructType` de sortie

Exceptions :

- `ASContextException` : si l'instance de schéma de sortie est `None`
- `InconsistentOutputDataModel` : les noms de champs et le type de stockage communs aux deux objets sont incohérents

Stocke l'emplacement de la sortie de génération du modèle dans le contexte Analytic Server et renvoie le contexte :

```
cxt.setModelContentFromPath(key, path, mimetype=None) : AnalyticServerContext
```

Le chemin peut être un chemin de répertoire qui doit utiliser l'API `cxt.createTemporaryFolder()` à générer, lorsque tout ce qui est sous le répertoire est packagé en tant que contenu de modèle.

Paramètres :

- `key (string)` : valeur de chaîne de clé
- `path (string)` : emplacement du chemin de la chaîne de sortie de génération du modèle
- `mimetype (string, optional)` : type MIME du contenu

Exceptions :

- `ModelOutputNotSupported` : en l'absence d'appel de cette API à partir de la fonction `pyspark:buildmodel`
- `KeyError` : si l'attribut de clé est `None` ou si la chaîne est vide

Stocke le contenu de génération du modèle, les métadonnées ou les autres attributs dans le contexte Analytic Server et renvoie le contexte :

```
cxt.setModelContentFromString(key, value, mimetype=None) : AnalyticServerContext
```

Paramètres :

- `key (string)` : valeur de chaîne de clé
- `value (string)` : valeur de la chaîne de métadonnées du modèle
- `mimetype (string, optional)` : type MIME du contenu

Exceptions :

- `ModelOutputNotSupported` : en l'absence d'appel de cette API à partir de la fonction `pyspark:buildmodel`
- `KeyError` : si l'attribut de clé est `None` ou si la chaîne est vide

Renvoie l'emplacement du dossier temporaire qui est géré par Analytic Server ; peut être utilisé pour stocker le contenu du modèle :

```
cxt.createTemporaryFolder() : string
```

Exception :

- `ModelOutputNotSupported` : en l'absence d'appel de cette API à partir de la fonction `pyspark:buildmodel`

Renvoie l'emplacement du modèle qui correspond à la clé d'entrée :

```
cxt.getModelContentToPath(key) : string
```

Paramètre :

- `key (string)` : valeur de chaîne de clé

Exceptions :

- `ModelInputNotSupported` : en l'absence d'appel de cette API à partir de la fonction `pyspark:applymodel`
- `KeyError` : si l'attribut de clé est `None` ou si la chaîne est vide
- `IncompatibleModelContentType` : si le type de contenu du modèle n'est pas un conteneur

Renvoie le contenu du modèle, les métadonnées du modèle ou d'autres attributs de ce dernier qui correspondent à la clé d'entrée :

`cxt.getModelContentToString(key) : string`

Paramètre :

- `key (string)` : valeur de chaîne de clé

Exceptions :

- `ModelInputNotSupported` : en l'absence d'appel de cette API à partir de la fonction `pyspark:applymodel`
- `KeyError` : si l'attribut de clé est `None` ou si la chaîne est vide ou si la clé n'existe pas
- `IncompatibleModelContentType` : si le type de contenu du modèle n'est pas cohérent

Renvoie le type MIME affecté à la clé d'entrée. Il renvoie `None` si le contenu spécifié n'est pas de type MIME :

`cxt.getModelContentMimeType(key) : string`

Paramètre :

- `key (string)` : valeur de chaîne de clé

Exceptions :

- `ModelInputNotSupported` : en l'absence d'appel de cette API à partir de la fonction `pyspark:applymodel`
- `KeyError` : si l'attribut de clé est `None` ou si la chaîne est vide ou si la clé n'existe pas

## Métadonnées de données

Cette section décrit comment configurer les attributs de modèle de données en fonction de `pyspark.sql.StructField`.

### Objets `spss.datamodel.Role`

Cette classe énumère les rôles valides pour chaque champ d'un modèle de données.

`BOTH` : indique que ce champ peut être un antécédent ou une conséquence.

`FREQWEIGHT` : indique que ce champ est utilisé comme pondération de fréquence ; n'est pas visible pour l'utilisateur.

`INPUT` : indique que ce champ est un prédicteur ou un antécédent.

`NONE` : indique que ce champ n'est pas utilisé directement durant la modélisation.

`TARGET` : indique que ce champ est une prédiction ou une conséquence.

`PARTITION` : indique que ce champ est utilisé pour identifier la partition de données.

`RECORDID` : indique que ce champ est utilisé pour identifier l'ID d'enregistrement.

`SPLIT` : indique que ce champ est utilisé pour fractionner les données.

### Objets `spss.datamodel.Measure`

Cette classe énumère les niveaux de mesure des champs dans un modèle de données.

`UNKNOWN` : indique que le type de mesure est inconnu.

`CONTINUOUS` : indique que le type de mesure est continu.

NOMINAL : indique que le type de mesure est nominal.

FLAG : indique que la valeur du champ est une des deux valeurs.

DISCRETE : indique que la valeur du champ doit être interprétée comme une collection de valeurs.

ORDINAL : indique que le type de mesure est ordinal.

TYPELESS : indique que le champ peut avoir toute valeur compatible avec son stockage.

## Objets `pyspark.sql.StructField`

Représente un champ dans un élément `StructType`. Un objet `StructField` se compose de quatre champs :

- `name` (`string`) : nom d'un `StructField`
- `dataType` (`pyspark.sql.DataType`) : type de données spécifique
- `nullable` (`bool`) : si les valeurs d'un `StructField` peuvent contenir des valeurs `None`
- `metadata` (`dictionary`) : dictionnaire Python utilisé pour stocker les attributs d'option

Vous pouvez utiliser l'instance de dictionnaire de métadonnées pour stocker l'attribut de mesure, de rôle ou de libellé du champ spécifique. Les mots clés de ces attributs sont :

- `measure` : mot clé de l'attribut `measure`
- `role` : mot clé de l'attribut `role`
- `displayLabel` : mot clé de l'attribut `label`

Exemple :

```
from spss.datamodel.Role import Role
from spss.datamodel.Measure import Measure
_metadata = {}
_metadata['measure'] = Measure.TYPELESS
_metadata['role'] = Role.NONE
_metadata['displayLabel'] = "field label description"
StructField("userName", StringType(), nullable=False,
metadata=_metadata)
```

## Date, heure, horodatage

Pour les opérations qui utilisent des données de type date, heure ou horodatage, la valeur est convertie en la valeur réelle basée sur la valeur 1970-01-01:00:00:00 (utilisation du Temps Universel Coordonné).

Pour la date, la valeur représente le nombre de jours, basé sur la valeur 1970-01-01 (utilisation du Temps Universel Coordonné).

Pour l'heure, la valeur représente le nombre de secondes sur 24 heures.

Pour l'horodatage, la valeur représente le nombre de secondes, basé sur la valeur 1970-01-01:00:00:00 (utilisation du Temps Universel Coordonné).

## Exceptions

Cette section décrit les instances d'exception possibles.

### Objets `MetadataException`

Sous-classe d'exception Python.

Cette exception est émise en cas d'erreur durant l'exécution de l'objet métadonnées.

## **Objets UnsupportedOperationException**

Sous-classe d'exception Python.

Cette exception est émise si l'opération spécifique n'autorise pas l'exécution.

## **Objets InconsistentOutputDataModel**

Sous-classe d'exception Python.

Cette exception est émise si `setSparkOutputSchema` et `setSparkOutputData` sont appelés mais que les noms de champs et les informations de type de stockage communs aux deux objets sont incohérents.

## **Objets IncompatibleModelContentType**

Sous-classe d'exception Python.

Cette exception est émise au cours des scénarios suivants :

- Utilisation de `setModelContentFormString` pour définir le modèle mais utilisation de `getModelContentToPath` pour extraire la valeur
- Utilisation de `setModelContentFormPath` pour définir le modèle mais utilisation de `getModelContentToString` pour extraire la valeur

## **Objets DataOutputNotSupported**

Sous-classe d'exception Python.

Cette exception est émise dans `setSparkOutputData` au cours d'une exécution gérée par la fonction `pyspark:buildmodel`.

## **Objets ModelInputNotSupported**

Sous-classe d'exception Python.

Cette exception n'est émise que si le script n'appelle pas les API `getModelContentPathByKey` et `getModelContentToString` dans la fonction `pyspark:applymodel`.

## **Objets ModelOutputNotSupported**

Sous-classe d'exception Python.

Cette exception n'est émise que si le script n'appelle pas les API `setModelContentFromPath` et `setModelContentFromString` dans la fonction `pyspark:buildmodel`.

## **Objets ASContextException**

Sous-classe d'exception Python.

Cette exception est émise en cas d'exception d'exécution inattendue.

## **Exemples**

Cette section contient des exemples de scriptage Python for Spark.

## Exemple de scriptage de base pour le traitement de données

```
import spss.pyspark.runtime
from pyspark.sql.types import *

cxt = spss.pyspark.runtime.getContext()

if cxt.isComputeDataModelOnly():
 _schema = cxt.getSparkInputSchema()
 cxt.setSparkOutputSchema(_schema)
else:
 _structType = cxt.getSparkInputSchema()
 df = cxt.getSparkInputData()
 _newDF = df.sample(False, 0.01, 1)
 cxt.setSparkOutputData(_newDF)
```

## Exemple de script de génération de modèle, en utilisant l'algorithme LinearRegressionWithSGD

```
from pyspark.context import SparkContext
from pyspark.sql.context import SQLContext
from pyspark.sql import Row
from pyspark.mllib.regression import
LabeledPoint, LinearRegressionWithSGD, LinearRegressionModel
from pyspark.mllib.linalg import DenseVector
import numpy
import json

import spss.pyspark.runtime
from spss.pyspark.exceptions import ASContextException

ascontext = spss.pyspark.runtime.getContext()
sc = ascontext.getSparkContext()
df = ascontext.getSparkInputData()

paramètres de champ et d'algorithme

target = '%target_field%'
predictors = ['%predictor_fields%']
num_iterations = '%num_iterations%'
prediction_field = "$LR-" + target

enregistrement du modèle de régression linéaire dans un chemin de système de fichiers

def save(model, sc, path):
 data =
sc.parallelize([json.dumps({"intercept":model.intercept,"weights":model.weights.tolist()})])
 data.saveAsTextFile(path)

impression des détails du modèle dans stdout

def dump(model,predictors):
 print(prediction_field+" = " + str(model.intercept))
 weights = model.weights.tolist()
 for i in range(0,len(predictors)):
 print("\t+ "+predictors[i]+"*"+ str(weights[i]))

vérification de l'existence des champs obligatoires dans les données d'entrée

input_field_names = [ty[0] for ty in df.dtypes[:]]
if target not in input_field_names:
 raise ASContextException("target field "+target+" not found")
for predictor in predictors:
 if predictor not in input_field_names:
 raise ASContextException("predictor field "+predictor+" not found")

définition de la fonction permettant de convertir des objets dataframe Row en mllib LabeledPoint

def row2LabeledPoint(target,predictors,row):
```



```

 pvals = []
 for predictor in predictors:
 pval = getattr(row,predictor)
 pvals.append(float(pval))
 tval = getattr(row,target)
 return LabeledPoint(float(tval),DenseVector(pvals))

conversion de trame de données en RDD contenant LabeledPoint

training_points = df.rdd.map(lambda row:
row2LabeledPoint(target,predictors,row))

génération du modèle

model = LinearRegressionWithSGD.train(training_points,num_iterations,intercept=True)

écriture d'une description textuelle du modèle dans stdout

dump(model,predictors)

enregistrement du modèle dans le système de fichiers et stockage dans le contenu du modèle de sortie

modelpath = ascontext.createTemporaryFolder()
save(model,sc,modelpath)
ascontext.setModelContentFromPath("model",modelpath)

```

## Exemple de script de scoring de modèle, en utilisant l'algorithme LinearRegressionWithSGD

```

import json
import spss.pyspark.runtime
from pyspark.sql import Row
from pyspark.mllib.regression import
LabeledPoint,LinearRegressionWithSGD, LinearRegressionModel
from pyspark.mllib.linalg import DenseVector
from pyspark.sql.context import SQLContext
import numpy
from pyspark.sql.types import DoubleType, StructField

ascontext = spss.pyspark.runtime.getContext()
sc = ascontext.getSparkContext()

prediction_field = "$LR-" + '%%target_field%%'
predictors = [%%predictor_fields%%]

calcul du schéma de sortie en ajoutant le champ de prédiction
outputSchema = ascontext.getSparkInputSchema()
outputSchema.fields.append(StructField(prediction_field,
DoubleType(), nullable=True))

exécution d'une prédiction basée sur un modèle de régression et un objet Dataframe Row (ligne de base de données)
renvoi d'une liste contenant les valeurs de ligne d'entrée et la valeur prévue
def predict(row,model,predictors,infields,prediction_field_name):
 pvals = []
 rdict = row.asDict()
 for predictor in predictors:
 pvals.append(float(rdict[predictor]))
 estimate = float(model.predict(pvals))
 result = []
 for field in infields:
 result.append(rdict[field])
 result.append(estimate)
 return result

chargement d'un modèle sérialisé à partir du système de fichiers

def load(sc, path):

```

```

js = sc.textFile(path).take(1)[0]
obj = json.loads(js)
weights = numpy.array(obj["weights"])
intercept = obj["intercept"]
return LinearRegressionModel(weights,intercept)

ascontext.setSparkOutputSchema(outputSchema)

if not ascontext.isComputeDataModelOnly():
 # scoring des données dans la trame de données d'entrée
 indf = ascontext.getSparkInputData()

 model_path = ascontext.getModelContentToPath("model")
 model = load(sc,model_path)

 # calcul des scores
 infield_names = [ty[0] for ty in indf.dtypes[:]]
 scores_rdd = indf.rdd.map(lambda row:predict(row,model,predictors,infield_names,prediction_field))

 # création d'une trame de données (DataFrame) de sortie contenant les scores
 sqlCtx = SQLContext(sc)
 outdf = sqlCtx.createDataFrame(scores_rdd,schema=outputSchema)

 # renvoi du DataFrame de sortie comme résultat
 ascontext.setSparkOutputData(outdf)

```

---

## Noeuds d'extension

En complément d'IBM SPSS Modeler et de ses possibilités d'exploration de données, les noeuds Extension permettent aux utilisateurs experts de créer leurs propres scripts R ou Python for Spark afin de procéder à des opérations de traitement des données, et de génération et d'évaluation de modèle.

### Noeud Exportation d'extension

Avec le noeud Exportation d'extension, vous pouvez exécuter des scripts R ou Python for Spark pour exporter des données.

#### Noeud Exportation d'extension - Onglet Syntaxe

Sélectionnez le type de syntaxe – **R** ou **Python for Spark**. Consultez les sections suivantes pour plus d'informations. Lorsque votre syntaxe est prête, vous pouvez cliquer sur **Exécuter** pour exécuter le noeud Exportation d'extension.

#### Syntaxe R

**Syntaxe R.** Permet d'entrer ou de coller la syntaxe de script R personnalisé en vue de l'analyse des données dans ce champ.

**Convertir les champs indicateurs.** Indique comment sont traités les champs indicateurs. Deux options sont disponibles : **Chaînes en facteur**, **Entiers et Réels en double** et **Valeurs logiques (True, False)**. Si vous sélectionnez **Valeurs logiques (True, False)**, les valeurs originales des champs indicateurs sont perdues. Par exemple, si un champ a les valeurs Mâle et Femelle, elles sont remplacées par True et False.

**Convertir les valeurs manquantes en valeur R 'non disponible' (NA).** Lorsque cette option est sélectionnée, toute valeur manquante est convertie en valeur R NA. La valeur NA est utilisée par R pour identifier les valeurs manquantes. Des fonctions R que vous utilisez peuvent comporter un argument par le biais duquel il est possible de contrôler le comportement des fonctions lorsque les données contiennent NA. Par exemple, la fonction peut vous permettre de choisir d'exclure automatiquement les enregistrements qui contiennent NA. Si cette option n'est pas sélectionnée, les valeurs manquantes sont transmises à R en l'état et peuvent entraîner des erreurs lors de l'exécution du script R.

**Convertir les champs date/heure en classes R avec contrôle spécial pour les fuseaux horaires.** Lorsque cette option est sélectionnée, les variables de format de date et de date/heure sont converties en objets R date/heure. Vous devez sélectionner l'une des options suivantes :

- **R POSIXct.** Les variables de format de date ou de date/heure sont converties en objets R POSIXct.
- **R POSIXlt (liste).** Les variables de format de date ou de date/heure sont converties en objets R POSIXlt.

**Remarque :** Les formats POSIX sont des options avancées. Utilisez-les uniquement si le script R spécifie que les champs date/heure sont traités de telle manière que ces formats sont requis. Les formats POSIX ne s'appliquent pas aux variables de format horaire.

## Syntaxe Python

**Syntaxe Python.** Permet d'entrer ou de coller la syntaxe de script Python personnalisé en vue de l'analyse des données dans ce champ. Pour plus d'informations relatives à Python for Spark, voir «Python for Spark», à la page 1 et «Scriptage avec Python for Spark», à la page 2.

## Noeud Exportation d'extension - Onglet Sortie de la console

L'onglet **Sortie de la console** contient les sorties reçues lorsque le script R ou le script Python for Spark de l'onglet **Syntaxe** est exécuté (par exemple, si un script R est utilisé, il affiche la sortie reçue de la console R lorsque le script R du champ **Syntaxe R** de l'onglet **Syntaxe** est exécuté). La sortie peut contenir des messages d'erreur ou d'avertissement R ou Python générés lors de l'exécution du script R ou Python. Cette sortie permet essentiellement de déboguer le script. L'onglet **Sortie de la console** contient également le script du champ **Syntaxe R** ou **Syntaxe Python**.

A chaque exécution du script Exportation d'extension, le contenu de l'onglet **Sortie de la console** est écrasé par la sortie reçue de la console R ou Python for Spark. La sortie ne peut pas être éditée.

## Publication de flux

La publication de flux a lieu directement depuis IBM SPSS Modeler à l'aide de l'un des noeuds d'exportation standard : Base de données, Fichier plat, Statistics Export, Extension Export, Data Collection Export, SAS Export, Excel et XML Export. Le type de noeud d'export détermine le format des résultats à écrire chaque fois que le flux publié est exécuté à l'aide d'IBM SPSS Modeler Solution Publisher Runtime ou d'une application externe. A titre d'exemple, pour enregistrer vos résultats dans une base de données chaque fois que le flux publié est exécuté, utilisez un noeud exportation de base de données.

### Pour publier un flux, procédez comme suit :

1. Ouvrez ou créez un flux, puis reliez un noeud d'export à la fin du flux.
2. Dans l'onglet Publication du noeud d'export, indiquez un nom racine pour les fichiers publiés (c'est-à-dire le nom de fichier auquel seront ajoutées les extensions .pim, .par et .xml).
3. Cliquez sur **Publier** pour publier le flux ou sélectionnez **Publier le flux** pour publier automatiquement le flux à chaque exécution du noeud.

**Nom publié.** Indiquez le nom racine des fichiers d'image et de paramètres publiés.

- Le **fichier image** (\*.pim) fournit toutes les informations nécessaires à l'exécution par Runtime du flux publié, exactement tel qu'il était au moment de l'exportation. Si vous ne pensez pas devoir changer de paramètres du flux (comme la source des données d'entrée ou le fichier des données de sortie), vous pouvez vous contenter de déployer le fichier image.
- Le **fichier de paramètres** (\*.par) contient des informations configurables sur les sources de données, les fichiers de sortie et les options d'exécution. Pour pouvoir contrôler l'entrée ou la sortie du flux sans republier ce dernier, vous avez besoin du fichier de paramètres et du fichier image.
- Le **fichier de métadonnées** (\*.xml) décrit les entrées et les sorties de l'image et de leurs modèles de données. Il est conçu pour être utilisé par des applications qui incorporent la bibliothèque d'exécution et qui ont besoin de connaître la structure des données d'entrée et de sortie.

**Remarque :** Ce fichier est uniquement généré si vous sélectionnez l'option **Publier les métadonnées**.

**Publication des paramètres.** Si nécessaire, vous pouvez inclure les paramètres de flux dans le fichier \*.par. Vous pouvez modifier les valeurs des paramètres de flux lorsque vous exécutez l'image soit en modifiant le fichier \*.par ou en utilisant l'API d'exécution.

Cette option active le bouton **Paramètres**. La boîte de dialogue Publier les paramètres s'ouvre lorsque vous cliquez sur le bouton.

Choisissez les paramètres que vous souhaitez inclure dans l'image publiée en sélectionnant l'option correspondante dans la colonne **Publication**.

**Exécution dans le flux.** Indique si le flux est publié automatiquement lorsque le noeud est exécuté.

- **Exporter les données.** Exécute le noeud d'export de manière standard, sans publier le flux. (Au départ, le noeud s'exécute dans IBM SPSS Modeler comme si IBM SPSS Modeler Solution Publisher n'était pas disponible). Si vous sélectionnez cette option, le flux n'est pas publié tant que vous ne cliquez pas sur **Publier** dans la boîte de dialogue de noeud d'export. Vous pouvez également publier le flux en cours à l'aide de l'outil Publier de la barre d'outils ou à l'aide d'un script.
- **Publier le flux.** Publie le flux en vue de le déployer à l'aide de IBM SPSS Modeler Solution Publisher. Sélectionnez cette option pour publier automatiquement le flux chaque fois qu'il est exécuté.

**Remarque :**

- Si vous comptez exécuter le flux publié avec de nouvelles données ou des données mises à jour, n'oubliez pas que l'ordre des champs du fichier d'entrée doit être identique à celui des champs du fichier d'entrée du noeud source spécifié dans le flux publié.
- Lorsque vous publiez des données vers des applications externes, vous pouvez être amené à filtrer les champs superflus ou à renommer des champs pour respecter les exigences de saisie. Pour ce faire, utilisez le noeud Filtrer avant le noeud d'export.

## Noeud Sortie d'extension

Si **Sortie à l'écran** est sélectionné dans l'onglet **Sortie** de la boîte de dialogue de noeud Sortie d'extension, la sortie à l'écran s'affiche dans une fenêtre de navigateur de sortie. La sortie est également ajoutée au gestionnaire des sorties. La fenêtre du navigateur de sortie comporte ses propres menus, lesquels vous permettent d'imprimer ou d'enregistrer la sortie, ou de l'exporter dans un autre format. Le menu **Edition** contient uniquement l'option **Copier**. Le navigateur de sortie du noeud Sortie d'extension comporte deux onglets : l'onglet **Sortie de texte** qui affiche une sortie de type texte et l'onglet **Sortie graphique** qui affiche des graphiques et des diagrammes.

Si **Sortie dans le fichier** est sélectionné dans l'onglet **Sortie** de la boîte de dialogue de noeud Sortie d'extension, la fenêtre du navigateur de sortie ne s'affiche pas suite à l'exécution réussie du noeud Sortie d'extension.

## Noeud Sortie d'extension - Onglet Syntaxe

Sélectionnez le type de syntaxe – **R** ou **Python for Spark**. Consultez les sections suivantes pour plus d'informations. Lorsque votre syntaxe est prête, vous pouvez cliquer sur **Exécuter** pour exécuter le noeud Sortie d'extension. Les objets de sortie sont ajoutés au gestionnaire de sorties ou, le cas échéant, au fichier indiqué dans le champ **Nom de fichier** de l'onglet **Sortie**.

## Syntaxe R

**Syntaxe R.** Permet d'entrer ou de coller la syntaxe de script R personnalisé en vue de l'analyse des données dans ce champ.

**Convertir les champs indicateurs.** Indique comment sont traités les champs indicateurs. Deux options sont disponibles : **Chaînes en facteur**, **Entiers et Réels en double** et **Valeurs logiques (True, False)**. Si

vous sélectionnez **Valeurs logiques (True, False)**, les valeurs originales des champs indicateurs sont perdues. Par exemple, si un champ a les valeurs Mâle et Femelle, elles sont remplacées par True et False.

**Convertir les valeurs manquantes en valeur R 'non disponible' (NA).** Lorsque cette option est sélectionnée, toute valeur manquante est convertie en valeur R NA. La valeur NA est utilisée par R pour identifier les valeurs manquantes. Des fonctions R que vous utilisez peuvent comporter un argument par le biais duquel il est possible de contrôler le comportement des fonctions lorsque les données contiennent NA. Par exemple, la fonction `peut` vous permet de choisir d'exclure automatiquement les enregistrements qui contiennent NA. Si cette option n'est pas sélectionnée, les valeurs manquantes sont transmises à R en l'état et peuvent entraîner des erreurs lors de l'exécution du script R.

**Convertir les champs date/heure en classes R avec contrôle spécial pour les fuseaux horaires.** Lorsque cette option est sélectionnée, les variables de format de date et de date/heure sont converties en objets R date/heure. Vous devez sélectionner l'une des options suivantes :

- **R POSIXct.** Les variables de format de date ou de date/heure sont converties en objets R POSIXct.
- **R POSIXlt (liste).** Les variables de format de date ou de date/heure sont converties en objets R POSIXlt.

**Remarque :** Les formats POSIX sont des options avancées. Utilisez-les uniquement si le script R spécifie que les champs date/heure sont traités de telle manière que ces formats sont requis. Les formats POSIX ne s'appliquent pas aux variables de format horaire.

## Syntaxe Python

**Syntaxe Python.** Permet d'entrer ou de coller la syntaxe de script Python personnalisé en vue de l'analyse des données dans ce champ. Pour plus d'informations relatives à Python for Spark, voir «Python for Spark», à la page 1 et «Scriptage avec Python for Spark», à la page 2.

## Noeud Sortie d'extension - Onglet Sortie de la console

L'onglet **Sortie de la console** contient les sorties reçues lorsque le script R ou le script Python for Spark de l'onglet **Syntaxe** est exécuté (par exemple, si un script R est utilisé, il affiche la sortie reçue de la console R lorsque le script R du champ **Syntaxe R** de l'onglet **Syntaxe** est exécuté). La sortie peut contenir des messages d'erreur ou d'avertissement R ou Python générés lors de l'exécution du script R ou Python. Cette sortie permet essentiellement de déboguer le script. L'onglet **Sortie de la console** contient également le script du champ **Syntaxe R** ou **Syntaxe Python**.

A chaque exécution du script **Sortie d'extension**, le contenu de l'onglet **Sortie de la console** est écrasé par la sortie reçue de la console R ou Python for Spark. La sortie ne peut pas être éditée.

## Noeud Sortie d'extension - Onglet Sortie

**Nom de la sortie** - Spécifie le nom de la sortie générée lorsque le noeud est exécuté. Lorsque **Auto** est sélectionné, le nom "Sortie R" ou "Sortie Python" est attribué automatiquement à la sortie selon le type de script. Si vous le souhaitez, vous pouvez choisir **Personnalisé** pour indiquer un autre nom.

**Sortie à l'écran.** Sélectionnez cette option pour générer et afficher la sortie dans une nouvelle fenêtre. La sortie est également ajoutée au gestionnaire des sorties.

**Sortie dans le fichier.** Sélectionnez cette option pour enregistrer la sortie dans un fichier. Les boutons radio **Graphique de sortie** et **Fichier de sortie** sont alors activés.

**Graphique de sortie.** Activé uniquement si **Sortie dans le fichier** est sélectionné. Sélectionnez cette option pour enregistrer des graphiques générés à la suite de l'exécution du noeud **Sortie d'extension** dans un fichier. Indiquez un nom de fichier à utiliser pour la sortie générée dans le champ **Nom de fichier**. Cliquez sur les points de suspension (...) pour choisir un fichier et un emplacement spécifiques. Indiquez le type du fichier dans la liste déroulante **Type de fichier**. Les types de fichier suivants sont disponibles :

- Objet de sortie (.cou)
- HTML (.html)

**Texte de sortie.** Activé uniquement si **Sortie dans le fichier** est sélectionné. Sélectionnez cette option pour enregistrer une sortie de texte générée suite à l'exécution du noeud Sortie d'extension dans un fichier. Indiquez un nom de fichier à utiliser pour la sortie générée dans le champ **Nom de fichier**. Cliquez sur les points de suspension (...) pour spécifier un fichier et un emplacement spécifiques. Indiquez le type de fichier dans la liste déroulante **Type de fichier**. Les types de fichier suivants sont disponibles :

- HTML (.html)
- Objet de sortie (.cou)
- Document texte (.txt)

### Navigateur Sortie d'extension

Si **Sortie à l'écran** est sélectionné dans l'onglet **Sortie** de la boîte de dialogue de noeud Sortie d'extension, la sortie à l'écran s'affiche dans une fenêtre de navigateur de sortie. La sortie est également ajoutée au gestionnaire des sorties. La fenêtre du navigateur de sortie comporte ses propres menus, lesquels vous permettent d'imprimer ou d'enregistrer la sortie, ou de l'exporter dans un autre format. Le menu **Edition** contient uniquement l'option **Copier**. Le navigateur de sortie du noeud Sortie d'extension contient deux onglets :

- L'onglet **Sortie de texte** affiche la sortie de texte
- L'onglet **Sortie de graphique** affiche les graphiques

Si **Sortie dans le fichier** est sélectionné dans l'onglet **Sortie** de la boîte de dialogue de noeud Sortie d'extension au lieu de **Sortie à l'écran**, la fenêtre du navigateur de sortie ne s'affiche pas suite à l'exécution réussie du noeud Sortie d'Extension.

**Navigateur Sortie d'extension - Onglet Sortie de texte :** L'onglet **Sortie de texte** affiche toutes les sorties de texte générées lors de l'exécution du script R ou Python for Spark sur l'onglet **Syntaxe** du noeud Sortie d'extension.

**Remarque :** Les messages d'erreur ou d'avertissement R ou Python for Spark renvoyés suite à l'exécution du script Sortie d'extension s'affichent toujours dans l'onglet **Sortie de la console** du noeud Sortie d'extension.

**Navigateur Sortie d'extension - Onglet Sortie de graphique :** L'onglet **Sortie de graphique** affiche tous les graphiques générés lors de l'exécution du script R ou Python for Spark sur l'onglet **Syntaxe** du noeud Sortie d'extension. Par exemple, si le script contient un appel à la fonction R `plot`, le graphique résultant s'affiche dans cet onglet.

### noeud Modèle d'extension

Avec le noeud Modèle d'extension, vous pouvez exécuter des scripts R ou Python for Spark pour générer et évaluer des modèles.

#### Noeud Modèle d'extension - Onglet Syntaxe

Sélectionnez le type de syntaxe – **R** ou **Python for Spark**. Ensuite, entrez ou collez votre syntaxe de script personnalisé dans l'un des champs suivants. Lorsque votre syntaxe est prête, vous pouvez cliquer sur **Exécuter** pour exécuter le noeud Modèle d'extension.

#### Syntaxe R

**Syntaxe de génération de modèle R.** Permet d'entrer ou de coller de la syntaxe de scripts R personnalisés en vue de la génération de modèle.

**Syntaxe de scoring du modèle R.** Permet d'entrer ou de coller de la syntaxe de scripts R personnalisés en vue du scoring de modèle.

### **Syntaxe Python for Spark**

**Syntaxe de génération du modèle Python.** Permet d'entrer ou de coller la syntaxe de scripts Python personnalisés de génération de modèle dans ce champ.

**Syntaxe d'évaluation du modèle Python.** Permet d'entrer ou de coller la syntaxe de scripts Python personnalisés en vue de l'évaluation du modèle dans ce champ.

Pour plus d'informations relatives à Python for Spark, voir «Python for Spark», à la page 1 et «Scriptage avec Python for Spark», à la page 2.

### **Noeud Modèle d'extension - Onglet Options de modèle**

**Nom du modèle.** Vous pouvez générer le nom du modèle automatiquement sur la base du champ cible ou ID (ou du type de modèle si aucun de ces champs n'est spécifié) ou spécifier un nom personnalisé.

### **Noeud Modèle d'extension - Onglet Sortie de la console**

L'onglet **Sortie de la console** contient les sorties reçues lorsque le script R ou le script Python for Spark de l'onglet **Syntaxe** est exécuté (par exemple, si un script R est utilisé, il affiche la sortie reçue de la console R lorsque le script R du champ **Syntaxe R** de l'onglet **Syntaxe** est exécuté). La sortie peut contenir des messages d'erreur ou d'avertissement R ou Python générés lors de l'exécution du script R ou Python. Cette sortie permet essentiellement de déboguer le script. L'onglet **Sortie de la console** contient également le script du champ **Syntaxe R** ou **Syntaxe Python**.

A chaque exécution du script Modèle d'extension, le contenu de l'onglet **Sortie de la console** est écrasé par la sortie reçue de la console R ou Python for Spark. La sortie ne peut pas être éditée.

### **Noeud Modèle d'extension - Onglet Sortie de texte**

L'onglet **Sortie de texte** est intégré au noeud Modèle d'extension si vous cochez la case **Afficher la sortie de texte R** de l'onglet **Options de modèle** dans la boîte de dialogue du noeud Modèle d'extension. Cet onglet peut uniquement afficher la sortie texte. Une sortie texte générée suite à l'exécution du script de génération de modèle R s'affiche dans cet onglet. Si vous exécutez à nouveau le script de génération de modèle sans spécifier un nom de modèle différent, le contenu de l'onglet **Sortie texte** de l'exécution précédente est écrasé. La sortie texte n'est pas éditable.

Si vous incluez un appel à la fonction R `sink` dans votre script, toute sortie produite après cette fonction est enregistrée dans le fichier spécifié et n'est pas affiché dans l'onglet **Sortie texte**.

**Remarque :** Les messages d'erreur ou d'avertissement R ou Python for Spark renvoyés suite à l'exécution du script de génération de modèle s'affichent toujours dans l'onglet **Sortie de la console** du noeud Modèle d'extension.

## **Nugget de modèle Extension**

Le nugget de modèle Extension est généré et placé dans la palette Modèles après l'exécution du noeud Modèle Extension, qui contient le script R ou Python for Spark définissant la génération de modèle et l'évaluation du modèle. Par défaut, le nugget Modèle d'extension contient le script utilisé pour l'évaluation du modèle, les options de lecture de données et les sorties de la console R ou Python for Spark. Le nugget Modèle d'extension peut éventuellement contenir d'autres formes de sortie, comme du texte ou des graphiques. Lorsque le nugget Modèle d'extension est généré et ajouté au canevas de flux, il est possible d'y connecter un noeud sortie. Ce dernier est alors utilisé de manière habituelle dans les flux IBM SPSS Modeler pour obtenir des informations sur les données et les modèles et pour exporter les données dans divers formats.

Pour utiliser ce noeud avec R, IBM SPSS Modeler - Essentials for R doit être installé. Pour les instructions d'installation et des informations sur la compatibilité, voir *IBM SPSS Modeler - Essentials for R: Installation Instructions*. Vous devez également disposer d'une version compatible de R sur votre ordinateur.

## Nugget de modèle Extension - Onglet Syntaxe

L'onglet **Syntaxe** figure toujours dans le nugget de modèle Extension.

**Syntaxe de scoring du modèle R.** Le script R utilisé pour l'évaluation du modèle apparaît dans ce champ. Par défaut, ce champ est activé mais non éditable. Pour éditer le script de scoring du modèle R, cliquez sur **Editer**.

**Syntaxe d'évaluation du modèle Python.** Si Python for Spark est utilisé, le script Python utilisé pour l'évaluation du modèle apparaît dans ce champ. Par défaut, ce champ est activé mais non éditable. Pour éditer le script d'évaluation du modèle Python, cliquez sur **Editer**.

**Editer.** Cliquez sur **Editer** pour passer le champ **Syntaxe d'évaluation du modèle Python** en mode éditable. Vous pouvez ensuite éditer votre script d'évaluation du modèle en renseignant le champ de syntaxe d'évaluation. Par exemple, vous pouvez éditer votre script d'évaluation du modèle s'il fait l'objet d'une erreur que vous avez identifiée après avoir exécuté le nugget de modèle Extension. Toutes les modifications apportées au script d'évaluation du modèle dans le nugget de modèle Extension sont perdues si vous régénérez le modèle en exécutant le noeud de modèle d'extension.

## Nugget de modèle d'extension - Options de modèle

L'onglet **Options de modèle** figure toujours dans le nugget de modèle Extension.

**Options de lecture de données.** Ces options s'appliquent uniquement à R, pas à Python for Spark. Ces options permettent de gérer les valeurs manquantes, les champs indicateurs et les variables liées au format de date et de date/heure.

- **Lecture des données par lots.** Si vous traitez un volume de données trop important pour la mémoire du moteur R, par exemple, utilisez cette option pour diviser les données en lots pouvant être envoyés et traités individuellement. Définissez le nombre maximum d'enregistrements de données à inclure dans chaque lot.

Pour le noeud Extension Transform et le nugget Extension Scoring, les données sont envoyées via le script R (par lots). C'est la raison pour laquelle, les scripts pour les noeuds de scoring de modèle et les noeuds de processus qui sont exécutés dans un environnement Hadoop ou Database ne doivent pas inclure des opérations couvrant ou combinant des lignes dans les données, comme le tri ou l'agrégation. Cette limitation est imposée pour faire en sorte que les données puissent être scindées dans un environnement Hadoop et au cours de l'exploration de base de données. Cette limitation ne s'applique pas si les scripts de scoring de modèle sont exécutés dans SPSS Modeler Server. Les noeuds Sortie d'extension et les noeuds Modèle d'extension ne sont pas concernés par cette limitation.

- **Convertir les champs indicateurs.** Indique comment sont traités les champs indicateurs. Deux options sont disponibles : **Chaînes en facteur**, **Entiers et Réels en double** et **Valeurs logiques (True, False)**. Si vous sélectionnez **Valeurs logiques (True, False)**, les valeurs originales des champs indicateurs sont perdues. Par exemple, si un champ a les valeurs Mâle et Femelle, elles sont remplacées par True et False.
- **Convertir les valeurs manquantes en valeur R 'non disponible' (NA).** Lorsque cette option est sélectionnée, toute valeur manquante est convertie en valeur R NA. La valeur NA est utilisée par R pour identifier les valeurs manquantes. Des fonctions R que vous utilisez peuvent comporter un argument par le biais duquel il est possible de contrôler le comportement des fonctions lorsque les données contiennent NA. Par exemple, la fonction peut vous permettre de choisir d'exclure automatiquement les enregistrements qui contiennent NA. Si cette option n'est pas sélectionnée, les valeurs manquantes sont transmises à R en l'état et peuvent entraîner des erreurs lors de l'exécution du script R.
- **Convertir les zones de date/heure en classes R avec contrôle spécial pour les fuseaux horaires** Lorsque cette option est sélectionnée, les variables avec des formats de date et date/heure sont converties en objets date/heure R. Vous devez sélectionner l'une des options suivantes :



- **R POSIXct.** Les variables de format de date ou de date/heure sont converties en objets R POSIXct.
- **R POSIXlt (liste).** Les variables de format de date ou de date/heure sont converties en objets R POSIXlt.

**Remarque :** Les formats POSIX sont des options avancées. Utilisez-les uniquement si le script R spécifie que les champs date/heure sont traités de telle manière que ces formats sont requis. Les formats POSIX ne s'appliquent pas aux variables de format horaire.

Les options qui sont sélectionnées pour les contrôles **Convertir les champs indicateurs**, **Convertir les valeurs manquantes en valeur R 'non disponible' (NA)** et **Convertir les champs de date/heure en classes R avec contrôle spécial pour les fuseaux horaires** ne sont pas reconnues lorsque le nuggé de modèle Extension est exécuté sur une base de données. Lorsque le noeud est exécuté sur une base de données, les valeurs par défaut de ces contrôles sont utilisées :

- **Convertir les champs indicateurs** est défini à la valeur **Chaînes en facteur, Entiers et Réels en double**.
- **Convertir les valeurs manquantes en valeur R 'non disponible' (NA)** est sélectionné.
- **Convertir les champs de date/heure en classes R avec contrôle spécial pour les fuseaux horaires** n'est pas sélectionné.

### **Nuggé de modèle Extension - Onglet Sortie de graphique**

L'onglet **Sortie de graphique** est intégré au nuggé de modèle Extension si vous cochez la case **Afficher les graphiques R en HTML** de l'onglet **Options de modèle** dans la boîte de dialogue du noeud Modèle d'extension. Les graphiques résultant de l'exécution du script R de génération de modèle s'affichent dans cet onglet. Par exemple, si le script contient un appel à la fonction R `plot`, le graphique résultant s'affiche dans cet onglet. Si vous exécutez à nouveau le script de génération de modèle sans spécifier un nom de modèle différent, le contenu de l'onglet **Sortie graphique** de l'exécution précédente est écrasé.

### **Nuggé de modèle Extension - Onglet Sortie de texte**

L'onglet **Sortie texte** est intégré au nuggé de modèle Extension si vous cochez la case **Afficher la sortie texte R** de l'onglet **Options de modèle** dans la boîte de dialogue du noeud Modèle d'extension. Cet onglet peut uniquement afficher la sortie texte. Une sortie texte générée suite à l'exécution du script de modèle d'extension s'affiche dans cet onglet. Si vous exécutez à nouveau le script de modèle d'extension sans spécifier un nom de modèle différent, le contenu de l'onglet **Sortie texte** de l'exécution précédente est écrasé. La sortie texte n'est pas éditable.

**Remarque :**

- Si vous incluez un appel à la fonction R `sink` dans votre script, toute sortie produite après cette fonction est enregistrée dans le fichier spécifié et n'est pas affiché dans l'onglet **Sortie texte**.
- Les messages d'erreur et d'avertissement renvoyés suite à l'exécution du script de modèle d'extension s'affichent toujours dans l'onglet **Sortie de la console** du noeud de modèle d'extension.

### **Nuggé de modèle Extension - Onglet Sortie de la console**

L'onglet **Sortie de la console** figure toujours dans le nuggé de modèle Extension. Il contient toutes les sorties reçues lorsque le script R ou le script Python for Spark de l'onglet **Syntaxe fonctionne** (par exemple, si un script R est utilisé, il affiche la sortie reçue de la console R lorsque le script R du champ **Syntaxe d'évaluation du modèle R** de l'onglet **Syntaxe** du nuggé de modèle Extension est exécuté). La sortie contient les messages d'erreur et d'avertissement R ou Python renvoyés au cours de l'exécution du script R ou Python, ainsi que la sortie texte de la console R. Cette sortie permet essentiellement de déboguer le script.

A chaque exécution du script d'évaluation du modèle, le contenu de l'onglet **Sortie de la console** est écrasé par la sortie reçue de la console R ou Python for Spark. La sortie de la console n'est pas éditable.

## Noeud de transformation d'extension

Avec le noeud de transformation d'extension, vous pouvez prendre les données d'un flux IBM SPSS Modeler et appliquer les transformations aux données à l'aide de script R ou Python for Spark. Une fois modifiées, les données sont renvoyées dans le flux à des fins de traitement, de génération de modèle et de scoring de modèle. Le noeud de transformation d'extension rend possible la transformation des données à l'aide d'algorithmes écrits en R ou Python for Spark et permet à l'utilisateur de développer des méthodes de transformation de données adaptées à une situation particulière.

Pour utiliser ce noeud avec R, IBM SPSS Modeler - Essentials for R doit être installé. Pour les instructions d'installation et des informations sur la compatibilité, voir *IBM SPSS Modeler - Essentials for R: Installation Instructions*. Vous devez également disposer d'une version compatible de R sur votre ordinateur.

### Noeud de transformation d'extension - Onglet Syntaxe

Sélectionnez le type de syntaxe – **R** ou **Python for Spark**. Consultez les sections suivantes pour plus d'informations. Lorsque votre syntaxe est prête, vous pouvez cliquer sur **Exécuter** pour exécuter le noeud de transformation d'extension.

#### Syntaxe R

**Syntaxe R.** Permet d'entrer ou de coller la syntaxe de script R personnalisé en vue de l'analyse des données dans ce champ.

**Convertir les champs indicateurs.** Indique comment sont traités les champs indicateurs. Deux options sont disponibles : **Chaînes en facteur**, **Entiers et Réels en double** et **Valeurs logiques (True, False)**. Si vous sélectionnez **Valeurs logiques (True, False)**, les valeurs originales des champs indicateurs sont perdues. Par exemple, si un champ a les valeurs Mâle et Femelle, elles sont remplacées par True et False.

**Convertir les valeurs manquantes en valeur R 'non disponible' (NA).** Lorsque cette option est sélectionnée, toute valeur manquante est convertie en valeur R NA. La valeur NA est utilisée par R pour identifier les valeurs manquantes. Des fonctions R que vous utilisez peuvent comporter un argument par le biais duquel il est possible de contrôler le comportement des fonctions lorsque les données contiennent NA. Par exemple, la fonction peut vous permettre de choisir d'exclure automatiquement les enregistrements qui contiennent NA. Si cette option n'est pas sélectionnée, les valeurs manquantes sont transmises à R en l'état et peuvent entraîner des erreurs lors de l'exécution du script R.

**Convertir les champs date/heure en classes R avec contrôle spécial pour les fuseaux horaires.** Lorsque cette option est sélectionnée, les variables de format de date et de date/heure sont converties en objets R date/heure. Vous devez sélectionner l'une des options suivantes :

- **R POSIXct.** Les variables de format de date ou de date/heure sont converties en objets R POSIXct.
- **R POSIXlt (liste).** Les variables de format de date ou de date/heure sont converties en objets R POSIXlt.

**Remarque :** Les formats POSIX sont des options avancées. Utilisez-les uniquement si le script R spécifie que les champs date/heure sont traités de telle manière que ces formats sont requis. Les formats POSIX ne s'appliquent pas aux variables de format horaire.

#### Syntaxe Python

**Syntaxe Python.** Permet d'entrer ou de coller la syntaxe de script Python personnalisé en vue de l'analyse des données dans ce champ. Pour plus d'informations relatives à Python for Spark, voir «Python for Spark», à la page 1 et «Scriptage avec Python for Spark», à la page 2.

### Noeud de transformation d'extension - Sortie de la console

L'onglet **Sortie de la console** contient les sorties reçues lorsque le script R ou le script Python for Spark de l'onglet Syntaxe est exécuté (par exemple, si un script R est utilisé, il affiche la sortie reçue de la

console R lorsque le script R du champ **Syntaxe R** de l'onglet **Syntaxe** est exécuté). La sortie peut contenir des messages d'erreur ou d'avertissement R ou Python générés lors de l'exécution du script R ou Python. Cette sortie permet essentiellement de déboguer le script. L'onglet **Sortie de la console** contient également le script du champ **Syntaxe R** ou **Syntaxe Python**.

A chaque exécution du script de transformation d'extension, le contenu de l'onglet **Sortie de la console** est écrasé par la sortie reçue de la console R ou Python for Spark. La sortie ne peut pas être éditée.

## Noeud Importation d'extension

Avec le noeud Importation d'extension, vous pouvez exécuter des scripts R ou Python for Spark pour importer des données.

### Noeud Importation d'extension - Onglet Syntaxe

Sélectionnez le type de syntaxe – **R** ou **Python for Spark**. Ensuite, entrez ou collez votre script personnalisé pour importer des données. Lorsque votre syntaxe est prête, vous pouvez cliquer sur **Exécuter** pour exécuter le noeud Importation d'extension.

### Noeud Importation d'extension - Onglet Sortie de la console

L'onglet **Sortie de la console** contient les sorties reçues lorsque le script R ou le script Python for Spark de l'onglet **Syntaxe** est exécuté (par exemple, si un script R est utilisé, il affiche la sortie reçue de la console R lorsque le script R du champ **Syntaxe R** de l'onglet **Syntaxe** est exécuté). La sortie peut contenir des messages d'erreur ou d'avertissement R ou Python générés lors de l'exécution du script R ou Python. Cette sortie permet essentiellement de déboguer le script. L'onglet **Sortie de la console** contient également le script du champ **Syntaxe R** ou **Syntaxe Python**.

A chaque exécution du script Importation d'extension, le contenu de l'onglet **Sortie de la console** est écrasé par la sortie reçue de la console R ou Python for Spark. La sortie ne peut pas être éditée.

## Filtrage ou modification du nom des champs

Vous pouvez renommer ou exclure des champs à tout stade d'un flux. Par exemple, en tant que chercheur en médecine, vous n'êtes peut-être pas intéressé par le niveau de potassium (données de niveau champ) des patients (données de niveau enregistrement). Vous pouvez donc filtrer le champ K (potassium) correspondant. Pour ce faire, vous pouvez utiliser un noeud Filtrer distinct ou un onglet Filtrer sur un noeud source ou de sortie. Cette fonctionnalité est identique quel que soit le noeud à partir duquel vous y accédez.

- A partir des noeuds source (fichier de variables, fichier fixe, fichier Statistics, XML ou importation d'extension, par exemple), vous pouvez renommer ou filtrer des champs au fur et à mesure de la lecture des données dans IBM SPSS Modeler.
- Le noeud Filtrer permet de renommer ou de filtrer les champs en tout point du flux.
- A partir des noeuds Export Statistics, Transformation Statistics, Modèle Statistics et Sortie Statistics, vous pouvez filtrer ou renommer les champs pour respecter les conventions de dénomination IBM SPSS Statistics. Pour plus d'informations, voir le rubrique Champs de changement de nom ou de filtrage de IBM SPSS Statistics.
- Vous pouvez utiliser l'onglet Filtre dans l'un des noeuds ci-dessus pour définir ou modifier des ensembles à réponses multiples. Pour plus d'informations, voir le rubrique Modification des ensembles à réponses multiples.
- Enfin, vous pouvez utiliser un noeud Filtrer pour mapper les champs entre un noeud source et un autre.



---

## Chapitre 2. Extensions

Les extensions sont des composants personnalisés qui étendent les fonctionnalités d'IBM SPSS Modeler. Les extensions sont regroupées dans des bundles d'extension (fichiers .mpe) et sont installées dans IBM SPSS Modeler. Elles peuvent être créées par n'importe quel utilisateur et partagées avec d'autres utilisateurs à l'aide du bundle d'extension associé.

Les utilitaires suivants, qui permettent d'utiliser les extensions, sont disponibles :

- Le «Hub d'extension», qui est accessible dans **Extensions > Hub d'extension**, est une interface destinée à la recherche, au téléchargement et à l'installation des extensions de la collection IBM SPSS Predictive Analytics sur GitHub. La boîte de dialogue Hub d'extension vous permet aussi d'afficher les détails des extensions installées sur votre ordinateur, de les mettre à jour et de les désinstaller.
- Pour installer un bundle d'extension stocké sur votre ordinateur local, sélectionnez **Extensions > Installer le bundle d'extension local**.
- Vous pouvez utiliser le générateur de boîtes de dialogue personnalisées pour les extensions pour créer une extension qui inclut une interface utilisateur; appelée boîte de dialogue de noeud personnalisée. Les boîtes de dialogue de noeud personnalisées génèrent un script R ou Python for Spark qui exécute les tâches associées à l'extension. Vous concevez le script généré lors de la conception de la boîte de dialogue personnalisée.

---

### Hub d'extension

La boîte de dialogue Hub d'extension permet d'effectuer les tâches suivantes :

- Explorer les extensions disponibles dans la collection IBM SPSS Predictive Analytics sur GitHub. Vous pouvez sélectionner des extension à installer maintenant, ou vous pouvez télécharger les extensions sélectionnées et les installer plus tard.
- Obtenir les mises à jour des extensions déjà installées sur l'ordinateur.
- Afficher des informations détaillées sur les extensions déjà installées sur l'ordinateur.
- Supprimer des extensions installées sur l'ordinateur.

Pour télécharger ou supprimer des extensions :

1. A partir du menu, choisissez **Extensions > Hub d'extension**
2. Sélectionnez les extensions que vous souhaitez télécharger ou supprimer et cliquez sur **OK**. Toutes les sélections effectuées dans les onglets Explorer et Installé sont traitées lorsque vous cliquez sur le bouton **OK**.

Par défaut, les extensions sélectionnées pour le téléchargement sont téléchargées et installées sur l'ordinateur. Dans l'onglet Paramètres, vous pouvez également choisir de télécharger les extensions sélectionnées dans un emplacement donné, sans les installer. Vous pourrez les installer ultérieurement en sélectionnant **Extensions > Installer le bundle d'extension local**. Pour Windows, vous pouvez installer une extension en cliquant deux fois sur le fichier du bundle d'extension.

#### Important :

- Pour Windows 7 et les versions ultérieures, l'installation d'une version mise à jour d'un bundle d'extension existant peut nécessiter l'exécution d'IBM SPSS Modeler avec des privilèges d'administrateur. Pour démarrer IBM SPSS Modeler avec des privilèges d'administrateur, cliquez avec le bouton droit de la souris sur l'icône d'IBM SPSS Modeler et choisissez **Exécuter en tant qu'administrateur**. Plus particulièrement, si vous recevez un message d'erreur vous informant de l'échec d'installation d'un ou de plusieurs bundles d'extension, faites une nouvelle tentative en tant qu'administrateur.

- Si vous vous connectez à Internet via un proxy, vous risquez de recevoir une erreur telle que "Certaines fonctions ne sont pas disponibles car aucune connexion Internet n'a été détectée" lorsque vous essayez d'ouvrir le Hub d'extension via l'option de menu **Extensions > Hub d'extension**. Pour résoudre ce problème, ajoutez les paramètres suivants aux options # JVM dans le fichier `jvm.cfg` qui se trouve dans le répertoire de configuration de votre installation SPSS Modeler. Enregistrez le fichier et redémarrez SPSS Modeler.

```
options, "-DproxyHost=proxyIP"
options, "-DproxyPort=proxyPort"
```

**Remarque :** Le contrat de licence que vous acceptez lors de l'installation d'une extension reste consultable à tout moment en cliquant sur le bouton **Plus d'infos...** de l'extension dans l'onglet Installé.

## Onglet Explorer

L'onglet Explorer contient toutes les extensions disponibles dans la collection IBM SPSS Predictive Analytics sur GitHub (<https://ibmpredictiveanalytics.github.io/>). Vous pouvez y sélectionner les nouvelles extensions à télécharger et à installer, et les mises à jour des extensions déjà installées sur l'ordinateur. L'onglet Explorer nécessite une connexion Internet.

- Pour chaque extension, le numéro de la dernière version et sa date sont affichés. Un bref récapitulatif de l'extension est aussi fourni. Le numéro de version des extensions qui sont déjà installées sur l'ordinateur est également indiqué.
- Vous pouvez afficher des informations détaillées relatives à une extension en cliquant sur **Plus d'infos**. Lorsqu'une mise à jour est disponible, **Plus d'infos** affiche les informations sur celle-ci.
- Pour connaître les conditions requises pour l'exécution d'une extension (par exemple si le IBM SPSS Modeler - Integration Plug-in for R est nécessaire), cliquez sur **Prérequis**. Lorsqu'une mise à jour est disponible, **Prérequis** affiche les informations sur celle-ci.

## Affiner par

Vous pouvez filtrer l'ensemble d'extensions affiché. Vous pouvez affiner l'ensemble en fonction de catégories générales, de la langue des extensions, du type d'organisation dont elles sont issues, ou de leur état. Dans chaque groupe, par exemple Catégorie, vous pouvez sélectionner plusieurs éléments pour filtrer la liste des extensions affichées. Vous pouvez aussi filtrer par termes de recherche. Les recherches ne sont pas sensibles à la casse, et l'astérisque (\*) est traité comme n'importe quel autre caractère. Il n'est pas associé à une recherche générique.

- Pour affiner la liste des extensions affichées, cliquez sur **Appliquer**. Appuyer sur la touche Entrée lorsque le curseur se trouve dans la zone **Rechercher** produit le même effet que cliquer sur **Appliquer**.
- Pour réinitialiser la liste de manière à afficher l'ensemble des extensions disponibles, supprimez le texte de la zone **Rechercher**, désélectionnez tout et cliquez sur **Appliquer**.

## Comment obtenir des plug-ins d'intégration

**Pour obtenir l'IBM SPSS Modeler - Integration Plug-in for R :**

Installez IBM SPSS Modeler - Essentials for R, disponible sur [https://github.com/IBMPredictiveAnalytics/R\\_Essentials\\_Modeler/releases/](https://github.com/IBMPredictiveAnalytics/R_Essentials_Modeler/releases/) ou dans la communauté IBM SPSS Predictive Analytics, sur <https://developer.ibm.com/predictiveanalytics/predictive-extensions/>. IBM SPSS Modeler - Essentials for R inclut IBM SPSS Modeler - Integration Plug-in for R. Essentials for R ne comprend pas le langage de programmation R. Avant d'installer IBM SPSS Modeler - Essentials for R, vous devez d'abord installer le langage de programmation R, s'il n'est pas encore installé. Il est disponible sur le site <http://www.r-project.org/>. Nous recommandons R 3.3.3.

Pour des informations détaillées sur l'installation et la configuration d', voir le wiki de présentation d'IBM SPSS Statistics - Integration Plug-in for R.

**Remarque :** Si vous installez Essentials for R sur un ordinateur ne disposant pas d'un accès Internet et que vous prévoyez d'utiliser les scripts de R inclus avec Essentials for R, vous devez vous procurer les packages R requis par ces scripts et les installer manuellement dans R. Pour déterminer les packages R requis pour un script de R, ouvrez la boîte de dialogue Hub d'extension (**Extensions > Hub d'extension**), cliquez sur l'onglet **Installé**, puis cliquez sur **Plus d'infos** pour l'extension souhaitée. Les packs R requis sont répertoriés dans la boîte de dialogue Détails de l'extension. Les modules R peuvent être obtenus à partir de n'importe quel site miroir R CRAN, disponibles sur <http://www.r-project.org/>. Prenez soin de vous procurer les versions des packs qui correspondent à votre version de R. Les packs spécifiques aux versions sont disponibles via les liens de la page "Contributed Packages" sur le site miroir CRAN.

## Onglet Installé

L'onglet Installé contient toutes les extensions qui sont installées sur l'ordinateur. Vous pouvez y sélectionner les mises à jour des extensions installées, disponibles dans la collection IBM SPSS Predictive Analytics sur GitHub et y supprimer des extensions. Une connexion Internet est nécessaire pour obtenir les mises à jour des extensions installées.

- Pour chaque extension, numéro de la version installée est affiché. Lorsqu'une connexion Internet est disponible, le numéro de la dernière version et sa date sont affichés. Un bref récapitulatif de l'extension est aussi fourni.
- Vous pouvez afficher des informations détaillées relatives à une extension en cliquant sur **Plus d'infos**. Lorsqu'une mise à jour est disponible, **Plus d'infos** affiche les informations sur celle-ci.
- Pour connaître les conditions requises pour l'exécution d'une extension (par exemple si le IBM SPSS Modeller - Integration Plug-in for R est nécessaire), cliquez sur **Prérequis**. Lorsqu'une mise à jour est disponible, **Prérequis** affiche les informations sur celle-ci.

## Affiner par

Vous pouvez filtrer l'ensemble d'extensions affiché. Vous pouvez affiner l'ensemble en fonction de catégories générales, de la langue des extensions, du type d'organisation dont elles sont issues, ou de leur état. Dans chaque groupe, par exemple Catégorie, vous pouvez sélectionner plusieurs éléments pour filtrer la liste des extensions affichées. Vous pouvez aussi filtrer par termes de recherche. Les recherches ne sont pas sensibles à la casse, et l'astérisque (\*) est traité comme n'importe quel autre caractère. Il n'est pas associé à une recherche générique.

- Pour affiner la liste des extensions affichées, cliquez sur **Appliquer**. Appuyer sur la touche Entrée lorsque le curseur se trouve dans la zone **Rechercher** produit le même effet que cliquer sur **Appliquer**.
- Pour réinitialiser la liste de manière à afficher l'ensemble des extensions disponibles, supprimez le texte de la zone **Rechercher**, désélectionnez tout et cliquez sur **Appliquer**.

## Extensions privées

Les extensions privées sont des extensions installées sur votre ordinateur, mais non disponibles dans la collection IBM SPSS Predictive Analytics sur GitHub. Les fonctionnalités permettant d'affiner l'ensemble des extensions affichées et de visualiser les prérequis pour l'exécution d'une extension ne sont pas disponibles pour les extensions privées.

**Remarque :** Lors de l'utilisation du hub d'extension sans connexion à Internet, certaines des fonctionnalités de l'onglet Installé risquent de ne pas être disponibles.

## Onglet Paramètres

L'onglet Paramètres définit si les extensions sélectionnées pour le téléchargement sont téléchargées et installées, ou téléchargées mais pas installées. Ce paramètre s'applique aux nouvelles extensions et aux mises à jour des extensions existantes. Vous pouvez choisir de télécharger les extensions sans les installer si vous les téléchargez dans le but de les diffuser à d'autres utilisateurs de votre organisation. Vous pouvez également choisir de télécharger les extensions sans les installer si vous ne disposez pas encore de la configuration requise, mais vous pensez l'obtenir bientôt.

Si vous choisissez de télécharger les extensions sans les installer, vous pourrez lancer l'installation ultérieurement en sélectionnant **Extensions > Installer le bundle d'extension local**. Pour Windows, vous pouvez installer une extension en cliquant deux fois sur le fichier du bundle d'extension.

## Détails de l'extension

La boîte de dialogue Détails de l'extension affiche les informations fournies par l'auteur de l'extension. En plus des informations nécessaires, tel que le récapitulatif et la version, il est possible que l'auteur ait inclus des URL vers des emplacements pertinents, tel que la page d'accueil de l'auteur. Si l'extension a été téléchargée depuis le Hub d'extension, elle comprend une licence qui peut être consultée en cliquant sur **Afficher la licence**.

**Noeuds personnalisés.** La table Noeuds personnalisés répertorie les boîtes de dialogue de noeuds personnalisés qui sont inclus dans l'extension.

**Remarque :** L'installation d'une extension qui contient une boîte de dialogue de noeud personnalisée peut nécessiter un redémarrage d' IBM SPSS Modeler pour afficher l'entrée de la boîte de dialogue de noeud dans la table Noeuds personnalisés.

**Dépendances :** Le groupe Dépendances répertorie les modules complémentaires nécessaires à l'exécution des composants inclus dans l'extension.

- **Integration Plug-In for R.** Les composants d'une extension peuvent nécessiter Integration Plug-in for R.
- **Packs R :** Répertorie tous les packs R nécessaires à l'extension. Pour plus d'informations, voir «Packs R nécessaires», à la page 27.

---

## Installation des bundles d'extension locaux

Pour installer un bundle d'extension stocké sur votre ordinateur local :

1. A partir des menus, sélectionnez :  
**Extensions > Installer le bundle d'extension local...**
2. Sélectionnez le bundle d'extension. Le type de fichier des bundles d'extensions est mpe.

**Important :** Pour les utilisateurs de Windows 7 et versions ultérieures de Windows, l'installation d'une version mise à jour d'un bundle d'extension existant peut nécessiter l'exécution d'IBM SPSS Modeler avec des privilèges d'administrateur. Pour démarrer IBM SPSS Modeler avec des privilèges d'administrateur, cliquez avec le bouton droit de la souris sur l'icône d'IBM SPSS Modeler et choisissez **Exécuter en tant qu'administrateur**. Plus particulièrement, si vous recevez un message d'erreur vous informant de l'échec d'installation d'un ou de plusieurs bundles d'extension, faites une nouvelle tentative en tant qu'administrateur.

## Emplacements de l'installation des extensions

Par défaut, les extensions sont installées dans un emplacement général auquel l'utilisateur a accès en écriture pour votre système d'exploitation.

Vous pouvez écraser l'emplacement par défaut en définissant un chemin à l'aide de la variable d'environnement `IBM_SPSS_MODELER_EXTENSIONS_PATH`. L'emplacement spécifié doit exister sur l'ordinateur cible. Après avoir défini `IBM_SPSS_MODELER_EXTENSIONS_PATH`, vous devez redémarrer IBM SPSS Modeler pour que les modifications soient prises en compte.

Pour créer une variable d'environnement sous Windows, à partir du Panneau de configuration :



## Windows 7

1. Sélectionnez Comptes utilisateurs.
2. Sélectionnez **Modifier mes variables d'environnement**.
3. Cliquez sur **Nouveau**, saisissez le nom de la variable d'environnement, (par exemple, IBM\_SPSS\_MODELER\_EXTENSIONS\_PATH) dans le champ **Nom de la variable** et entrez le(s) chemin(s) dans le champ Valeur de la variable.

## Windows 8 et versions ultérieures

1. Sélectionnez Système.
2. Sélectionnez l'onglet Avancé et cliquez sur **Variables d'environnement**. L'onglet Avancé est accessible à partir des Paramètres système avancés.
3. Dans la section Variables utilisateur, cliquez sur **Nouveau**, saisissez le nom de la variable d'environnement (par exemple, IBM\_SPSS\_MODELER\_EXTENSIONS\_PATH) dans le champ **Nom de variable** et entrez le(s) chemin(s) dans le champ Valeur de la variable.

**Important :** Pour les utilisateurs de Windows 7 et versions ultérieures de Windows, l'installation d'une version mise à jour d'un bundle d'extension existant peut nécessiter l'exécution d'IBM SPSS Modeler avec des privilèges d'administrateur. Pour démarrer IBM SPSS Modeler avec des privilèges d'administrateur, cliquez avec le bouton droit de la souris sur l'icône d'IBM SPSS Modeler et choisissez **Exécuter en tant qu'administrateur**. Plus particulièrement, si vous recevez un message d'erreur vous informant de l'échec d'installation d'un ou de plusieurs bundles d'extension, faites une nouvelle tentative en tant qu'administrateur.

## Packs R nécessaires

Si vous ne disposez pas d'un accès Internet, vous devrez obtenir les packs R requis pour une extension particulière, qui ne se trouvent pas sur votre ordinateur, auprès d'une personne disposant d'un tel accès. Vous pouvez consulter la liste des packs R requis dans la boîte de dialogue Détails de l'extension, une fois l'extension installée. Pour plus d'informations, reportez-vous à la rubrique «Détails de l'extension», à la page 26. Les packs peuvent être téléchargés depuis et installés dans R. Pour plus de détails, reportez-vous au guide *Installation et administration de R* fourni avec R.

**Remarque :** Pour les utilisateurs d'UNIX (y compris Linux), les packs sont téléchargés sous la forme de sources puis compilés. Ceci nécessite que vous disposiez des outils appropriés installés sur votre machine. Reportez-vous au guide *Installation et administration de R* pour plus de détails. En particulier, les utilisateurs de Debian doivent installer le pack r-base-dev à partir de apt-get install r-base-dev.

---

## Création et gestion des noeuds personnalisés

Le générateur de boîtes de dialogue personnalisées pour les extensions crée des noeuds à utiliser dans les flux SPSS Modeler.

A l'aide du générateur de boîtes de dialogue personnalisées pour les extensions vous pouvez :

- Créer une boîte de dialogue de noeud personnalisée pour exécuter un noeud implémenté dans R ou Apache Spark (via Python). Pour plus d'informations, voir «Création du modèle de script», à la page 31.
- Ouvrir un fichier contenant la spécification d'une boîte de dialogue de noeud personnalisée, éventuellement créée par un autre utilisateur, et ajouter celle-ci à votre installation IBM SPSS Modeler en y apportant le cas échéant vos propres modifications.
- Enregistrer la spécification d'une boîte de dialogue de noeud personnalisée afin que d'autres utilisateurs puissent l'ajouter à leur installation IBM SPSS Modeler.

- Créez des noeuds personnalisés et écrivez des scripts Python for Spark pour lire les données de votre source de données, où qu'elle se trouve, et écrire des données dans tout format pris en charge par Apache Spark. Pour plus d'informations, voir «Importation et exportation de données à l'aide de Python for Spark», à la page 59.
- Créez des noeuds personnalisés et écrivez des scripts R pour lire les données de votre source de données, où qu'elle se trouve, et écrire des données dans tout format pris en charge par R. Pour plus d'informations, voir «Importation et exportation de données à l'aide de R», à la page 59.

Dans le générateur de boîtes de dialogue personnalisées pour les extensions, vous créez ou modifiez des boîtes de dialogue de noeud personnalisées dans des extensions. Lorsque vous ouvrez le générateur de boîtes de dialogue personnalisées pour les extensions, une nouvelle extension contenant une boîte de dialogue de noeud personnalisée vide est créée. Lorsque vous enregistrez ou installez des boîtes de dialogue de noeud personnalisées à partir du générateur de boîtes de dialogue personnalisées pour les extensions, elles sont enregistrées ou installées comme partie intégrante d'une extension.

**Remarque :**

- Vous ne pouvez pas créer votre propre version d'une boîte de dialogue de noeud pour un noeud IBM SPSS Modeler standard.
- Le scriptage n'est pas pris en charge pour les noeuds créés avec le générateur de boîtes de dialogue personnalisées et notamment les noeuds R du générateur de boîtes de dialogue personnalisées et les noeuds Python du générateur de boîtes de dialogue personnalisées.

## **Comment lancer le générateur de boîtes de dialogue personnalisées pour les extensions**

A partir des menus, choisissez **Extensions > Générateur de boîtes de dialogue de noeud personnalisées**

**Remarque :**

- Les noeuds Python dépendent de l'environnement Spark.
- Les scripts Python doivent utiliser l'API Spark car les données sont présentées sous la forme d'un Spark DataFrame.
- Les anciens noeuds créés dans la version 17.1 ne seront toujours exécutés que sur IBM SPSS Analytic Server (les données proviennent d'un noeud source IBM SPSS Analytic Server et n'ont pas été extraites dans IBM SPSS Modeler Server). Les nouveaux noeuds Python et Générateur de boîtes de dialogue personnalisées créés dans la version 18.0 ou une version ultérieure peuvent être exécutés sur IBM SPSS Modeler Server.
- Lors de l'installation de Python, assurez-vous que tous les utilisateurs disposent de droits d'accès à l'installation de Python.
- Si vous souhaitez utiliser Machine Learning Library (MLlib), vous devez installer une version de Python qui inclut NumPy. Vous devez ensuite configurer IBM SPSS Modeler Server (ou le serveur local dans IBM SPSS Modeler Client) pour utiliser votre installation Python. Pour plus d'informations, voir «Scriptage avec Python for Spark», à la page 2.

## **Présentation du générateur de boîtes de dialogue personnalisées**

### **Canevas de boîte de dialogue**

Le canevas de boîte de dialogue est la zone du générateur de boîtes de dialogue personnalisées dans laquelle vous concevez la présentation de la boîte de dialogue de noeud.

### **Sous-fenêtre des propriétés**

La sous-fenêtre des propriétés est la zone du générateur de boîtes de dialogue personnalisées dans laquelle vous spécifiez les propriétés des contrôles qui composent la boîte de dialogue de noeud , ainsi

que les propriétés de la boîte de dialogue elle-même, comme le type du noeud.

## Palette d'outils

La palette d'outils fournit l'ensemble des contrôles qui peuvent être inclus dans une boîte de dialogue de noeud personnalisée. Vous pouvez afficher ou masquer la palette d'outils en sélectionnant Palette d'outils dans le menu **Affichage**.

## Modèle de script

Le modèle de script définit le script R ou Python for Spark généré par la boîte de dialogue de noeud. Vous pouvez déplacer la sous-fenêtre Modèle de script vers une fenêtre distincte en cliquant sur **Move to New Window** (Déplacer vers une nouvelle fenêtre). Pour ramener une fenêtre Modèle de script distincte dans le générateur de boîtes de dialogue personnalisées, cliquez sur **Restore to Main Window** (Restaurer dans la fenêtre principale).

## Création d'une boîte de dialogue de noeud personnalisée

Les étapes de base de la génération d'une boîte de dialogue de noeud personnalisée sont les suivantes :

1. Spécifiez les propriétés de la boîte de dialogue de noeud elle-même, comme le titre qui apparaît lors du lancement de celle-ci et l'emplacement du nouveau noeud dans les palettes IBM SPSS Modeler. Pour plus d'informations, voir «Propriétés des boîtes de dialogue».
2. Spécifiez les contrôles, comme les sélecteurs de champ et les cases à cocher, qui composent la boîte de dialogue de noeud et ses sous-boîtes de dialogue éventuelles. Pour plus d'informations, voir «Types de contrôle», à la page 32.
3. Créez le modèle de script qui spécifie le code R ou Python for Spark généré par la boîte de dialogue de noeud. Pour plus d'informations, voir la rubrique «Création du modèle de script», à la page 31.
4. Spécifiez les propriétés de l'extension qui contient votre boîte de dialogue de noeud. Pour plus d'informations, voir la rubrique «Propriétés d'extension», à la page 52.
5. Installez l'extension qui contient la boîte de dialogue de noeud dans IBM SPSS Modeler et/ou enregistrez l'extension dans un fichier de bundle d'extension (.mpe). Voir la rubrique «Gestion des boîtes de dialogue de noeud personnalisées», à la page 54 pour plus d'informations.

Vous pouvez afficher un aperçu de votre boîte de dialogue de noeud au cours de sa création. Pour plus d'informations, voir «Prévisualisation d'une boîte de dialogue de noeud personnalisée», à la page 32.

## Propriétés des boîtes de dialogue

La fenêtre Générateur de boîtes de dialogue personnalisées affiche les propriétés de la boîte de dialogue de noeud et du contrôle d'interface utilisateur sélectionné. Pour afficher et définir les Propriétés de boîte de dialogue, cliquez sur le canevas dans une zone située en dehors des contrôles. Sans contrôle sur le canevas, les propriétés de boîte de dialogue sont toujours visibles.

**Nom de la boîte de dialogue.** La propriété Nom de la boîte de dialogue est requise et indique un nom unique à associer à la boîte de dialogue de noeud. Pour réduire les risques de conflits de nom, il peut s'avérer nécessaire d'ajouter au nom un préfixe correspondant à un identificateur de l'organisation, comme une adresse URL.

**Titre.** La propriété Titre indique le texte à afficher dans la barre de titre de la boîte de dialogue de noeud.

**Fichier d'aide.** La propriété Fichier d'aide est facultative et indique le chemin d'accès à un fichier d'aide de la boîte de dialogue de noeud. Il s'agit du fichier qui est lancé lorsque l'utilisateur clique sur le bouton **Aide** dans la boîte de dialogue. Les fichiers d'aide doivent être au format HTML. Une copie du fichier

d'aide indiqué est incluse avec la spécification de la boîte de dialogue de noeud lorsque la boîte de dialogue de noeud est installée ou enregistrée. Le bouton Aide de la boîte de dialogue d'exécution est masqué s'il n'existe pas de fichier d'aide associé.

- Les versions localisées du fichier d'aide qui se trouvent dans le même répertoire que le fichier d'aide sont automatiquement ajoutées à la boîte de dialogue de noeud lorsque vous ajoutez le fichier d'aide. Les versions localisées du fichier d'aide s'intitulent <Fichier d'aide>\_<identificateur\_langue>.htm. Pour plus d'informations, voir la rubrique «Création de versions localisées des boîtes de dialogue de noeud personnalisées», à la page 57.
- Les fichiers pris en charge, comme les fichiers d'image et les feuilles de style, peuvent être ajoutés à la boîte de dialogue de noeud en enregistrant la boîte de dialogue de noeud au préalable. Vous ajoutez ensuite manuellement les fichiers d'aide au fichier de la boîte de dialogue de noeud (.cfe). Pour plus d'informations sur l'accès aux fichiers de la boîte de dialogue de noeud et à leur modification manuelle, reportez-vous à la section intitulée "Pour localiser des chaînes de boîte de dialogue" de la rubrique «Création de versions localisées des boîtes de dialogue de noeud personnalisées», à la page 57.

**Type de script.** Indique le type du script pouvant être utilisé pour générer le modèle de script. Dans IBM SPSS Modeler, seul le script R ou Python for Spark peut être utilisé.

**Score issu du modèle.** Indique si le modèle généré à l'aide du script de génération de modèle doit être utilisé pour le scoring.

**Type de noeud.** Indique le type du noeud qui sera créé lors de l'installation de votre boîte de dialogue de noeud.

**Palette.** Spécifie la palette dans laquelle le nouveau noeud sera ajouté lors de la création de votre boîte de dialogue de noeud.

**Icône de noeud.** Cliquez sur le bouton des points de suspension (...) pour sélectionner une image à utiliser comme icône du nouveau noeud. L'image que vous choisissez doit être un fichier .gif.

## Présentation des contrôles dans le canevas de boîte de dialogue

Vous ajoutez des contrôles à une boîte de dialogue de noeud personnalisée en les faisant glisser à partir de la palette d'outils dans le canevas de boîte de dialogue. Pour assurer la cohérence dans les boîtes de dialogue de noeud intégrées, le canevas de boîte de dialogue est divisé en quatre colonnes fonctionnelles dans lesquelles vous pouvez placer des contrôles.

- La première colonne (la plus à gauche) est principalement destinée à un contrôle Sélecteur de champ.
- Les boutons situés sous les boîtes de dialogue doivent se trouver dans la colonne la plus à droite (par exemple, la troisième colonne si on n'utilise que trois colonnes) et aucun autre contrôle ne peut se trouver dans la même colonne que les boutons situés sous les boîtes de dialogue (Sub-dialog). La quatrième colonne ne peut contenir que des boutons de type Sub-dialog.

Bien que cela n'apparaisse pas dans le canevas de boîte de dialogue, lorsque la boîte de dialogue de noeud est installée dans IBM SPSS Modeler, les boutons appropriés sont ajoutés à la boîte de dialogue (par exemple **OK**, **Annuler**, **Appliquer**, **Réinitialiser** et, le cas échéant, **Aide** et **Exécuter**). La présence et les emplacements de ces boutons sont automatiques. Toutefois, le bouton **Aide** est masqué si aucun fichier d'aide n'est associé à la boîte de dialogue de noeud (comme indiqué par la propriété Fichier d'aide dans Propriétés de la boîte de dialogue).

Vous pouvez modifier l'ordre vertical des contrôles dans une colonne en les faisant glisser vers le haut ou vers le bas, mais leur position exacte est déterminée automatiquement. Lors de l'exécution, les contrôles sont redimensionnés de manière appropriée lorsque la boîte de dialogue elle-même est redimensionnée. Les contrôles comme les sélecteurs de champ s'agrandissent automatiquement pour remplir l'espace disponible sous eux.

## Création du modèle de script

Le modèle de script indique le script R ou Python for Spark qui sera généré par la boîte de dialogue de noeud personnalisée. Une seule boîte de dialogue de noeud personnalisée peut être utilisée pour spécifier une ou plusieurs opérations devant s'exécuter les unes après les autres.

Le modèle de script peut être composé de *texte statique*. Le texte statique est différent du contrôle de texte statique. Il s'agit du code R ou Python for Spark qui est toujours généré lorsque le noeud est exécuté. Par exemple, les noms de commande et les spécifications de sous-commande qui ne dépendent pas des entrées utilisateur sont du texte statique. Le modèle de script peut également être composé d'identificateurs de contrôle qui sont remplacés à chaque exécution par les valeurs des contrôles de boîtes de dialogue de noeud personnalisées. Par exemple, l'ensemble des champs spécifiés dans un sélecteur de champ est représenté par l'identificateur de contrôle du contrôle de sélecteur de champ.

### Pour créer le modèle de script :

1. Pour le texte statique qui ne dépend pas des valeurs spécifiées par l'utilisateur, entrez le script R ou Python for Spark voulu, par exemple le champ **Syntaxe de génération du modèle R** du noeud de création R.
2. Ajoutez les identificateurs de contrôle au format `%Identifieur%` aux emplacements où vous voulez insérer le script R ou Python for Spark généré par les contrôles, où `Identifieur` correspond à la valeur de la propriété `Identifieur` du contrôle.
  - Vous pouvez insérer un identificateur de contrôle en sélectionnant une ligne du tableau des identificateurs, en cliquant avec le bouton droit et en sélectionnant **Ajouter au modèle de script**. Vous pouvez également insérer un identificateur de contrôle en cliquant avec le bouton droit sur le contrôle dans le canevas et en sélectionnant **Ajouter au modèle de script**.
  - Vous pouvez aussi effectuer une sélection à partir d'une liste des identificateurs de contrôle disponibles en appuyant sur `Ctrl+barre d'espace`. La liste contient les identificateurs de contrôle suivis des éléments disponibles avec la fonction d'exécution automatique du script.

Si vous saisissez des identificateurs, conservez les espaces car tous les espaces des identificateurs sont pris en compte.

Lors de l'exécution, et pour tous les contrôles autres que les cases à cocher, les groupes de cases à cocher et le contrôle de texte statique, chaque identificateur est remplacé par la valeur en cours de la propriété du **Script** du contrôle associé. Si lors de l'exécution, le contrôle est vide, aucun script n'est généré. Pour les cases à cocher et les groupes de cases à cocher, l'identificateur est remplacé par la valeur en cours de la propriété `Script R vérifié` ou `Script R non vérifié` du contrôle associé en fonction de l'état en cours du contrôle, `vérifié` ou `non vérifié`. Pour plus d'informations, voir «Types de contrôle», à la page 32.

### Exemple : inclusion de valeurs d'exécution dans le modèle de script R

Dans cet exemple, la boîte de dialogue de noeud personnalisée génère et exécute le script R pour créer et évaluer un modèle de régression linéaire en utilisant un appel à la fonction R `lm` avec la signature indiquée ci-dessous.

```
lm(formule,données)
```

- `formule` correspond à une expression, comme `Na~Age`, où `Na` correspond à la zone cible du modèle et où la zone d'entrée du modèle correspond à `Age`.
- `données` correspond à un cadre de données contenant les valeurs des champs spécifiés dans la formule.

Prenons l'exemple d'une boîte de dialogue de noeud personnalisée comportant un contrôle de sélecteur de champ unique qui permet à l'utilisateur de choisir la zone d'entrée du modèle linéaire. Le modèle de script permettant de générer et d'exécuter le script R qui génère le modèle est entré dans l'onglet **Script** et peut se présenter comme suit :

```
modelerModel <- lm(Na~%input%,data=modelerData)
```

- `%%input%` correspond à la valeur de la propriété Identificateur du contrôleur de sélecteur de champ. Lors de l'exécution, elle sera remplacée par la valeur en cours de la propriété du **Script** du contrôle.
- Le fait de définir la propriété du **Script** du contrôle de sélecteur de champ comme étant `%%ThisValue%` indique que lors de l'exécution la valeur en cours de la propriété sera la valeur du contrôle qui est le champ choisi dans le sélecteur de champ.

Supposons que l'utilisateur de la boîte de dialogue de noeud personnalisée sélectionne le champ Age comme zone d'entrée du modèle. Le script R suivant est alors généré par la boîte de dialogue de noeud :

```
modelerModel<-lm(Na~Age,data=modelerData)
```

Le modèle de script permettant de générer et d'exécuter le script R qui évalue le modèle est entré dans l'onglet **Script de score** et peut se présenter comme suit :

```
result<-predict(modelerModel,newdata=modelerData)
var1 <-c(fieldName="predicted", fieldLabel="",fieldStorage="real",fieldMeasure="",fieldFormat="",
fieldRole="")
modelerDataModel<-data.frame(modelerDataModel,var1)
```

Ce script R ne dépend pas des valeurs spécifiées par l'utilisateur mais uniquement du modèle créé à l'aide du script R de création de modèle. Ainsi, le script R de scoring de modèle est entré comme il le serait dans le champ **Syntaxe de scoring du modèle R** du noeud de création R.

## Prévisualisation d'une boîte de dialogue de noeud personnalisée

Vous pouvez afficher un aperçu de la boîte de dialogue de noeud actuellement ouverte dans le générateur de boîtes de dialogue personnalisées. La boîte de dialogue apparaît et fonctionne comme si elle était exécutée à partir d'un noeud dans IBM SPSS Modeler.

- Les sélecteurs de champ sont remplis avec des champs factices.
- Le bouton **OK** ferme l'aperçu.
- Si un fichier d'aide est spécifié, le bouton **Aide** est activé et ouvre le fichier indiqué. Sinon, le bouton d'aide est désactivé lors de l'affichage de l'aperçu et masqué lors de l'exécution de la boîte de dialogue réelle.

Pour afficher l'aperçu d'une boîte de dialogue de noeud personnalisée, dans les menus du générateur de boîtes de dialogue personnalisées, choisissez **Fichier > Aperçu de la boîte de dialogue**.

## Types de contrôle

La palette d'outils fournit tous les contrôles standard pouvant s'avérer nécessaires dans une boîte de dialogue de noeud personnalisée.

- **Sélecteur de champ** : Liste de tous les champs à partir de l'ensemble de données actif. Pour plus d'informations, voir la rubrique «Sélecteur de champs», à la page 33.
- **Case à cocher** : Case à cocher unique. Pour plus d'informations, voir «Case à cocher», à la page 34.
- **Zone de liste déroulante** : Zone de liste déroulante pour la création de listes déroulantes. Pour plus d'informations, voir «Zone de liste déroulante», à la page 35.
- **Zone de liste** : Zone de liste pour la création d'une liste de sélection unique ou de listes de sélection multiples. Pour plus d'informations, voir «Zone de liste déroulante», à la page 35.
- **Contrôle de texte** : Zone de texte qui accepte le texte arbitraire comme entrée. Pour plus d'informations, voir «Contrôle du texte», à la page 38.
- **Contrôle des nombres** : Zone de texte dont les entrées sont limitées aux valeurs numériques. Pour plus d'informations, voir «Contrôle des nombres», à la page 39.
- **Contrôle de date** : Contrôle spinner permettant de spécifier les valeurs de date/heure (date, heure et horodatage). Pour plus d'informations, voir «Date», à la page 40.
- **Texte sécurisé** : Zone de texte qui masque l'entrée utilisateur avec des astérisques. Pour plus d'informations, voir «Texte sécurisé», à la page 41.

- **Contrôle de texte statique** : Contrôle pour l'affichage de texte statique. Pour plus d'informations, voir «Contrôle de texte statique», à la page 42.
- **Sélecteur de couleur** : Contrôle permettant de spécifier une couleur et de générer la valeur RGB associée. Pour plus d'informations, voir «Sélecteur de couleur», à la page 43.
- **Contrôle de table** : Table avec un nombre fixe de colonnes et un nombre variable de lignes ajoutés lors de l'exécution. Pour plus d'informations, voir «Table», à la page 43.
- **Groupe d'éléments** : Conteneur pour le regroupement d'un ensemble de contrôles, comme un ensemble de cases à cocher. Pour plus d'informations, voir «Groupe d'éléments», à la page 46.
- **Groupe de boutons radio** : Groupe de boutons radio. Pour plus d'informations, voir «Groupe de boutons radio», à la page 46.
- **Groupes de cases à cocher** : Conteneur d'un ensemble de contrôles pouvant être activés ou désactivés en tant que groupe par une case à cocher unique. Pour plus d'informations, voir «Groupes de cases à cocher», à la page 47.
- **Navigateur de fichiers** : Contrôle de navigation dans le système de fichiers pour l'ouverture ou l'enregistrement d'un fichier. Pour plus d'informations, voir «Navigateur de fichiers», à la page 48.
- **Onglet** : Onglet unique. Pour plus d'informations, voir «Onglet», à la page 50.
- **Bouton de la sous-boîte de dialogue** : Bouton de lancement d'une sous-boîte de dialogue. Pour plus d'informations, voir «Bouton de la sous-boîte de dialogue», à la page 50.

## Sélecteur de champs

Le contrôle Sélecteur de champ (Field Chooser) affiche la liste des champs dont dispose l'utilisateur final de la boîte de dialogue de noeud. Vous pouvez enfin afficher tous les champs du jeu de données actif (par défaut) ou filtrer la liste en fonction du type et du niveau de mesure (par exemple, champs numériques ayant un niveau de mesure d'échelle). Vous pouvez également spécifier tout autre contrôle Sélecteur de champ comme source des champs pour le sélecteur de champ en cours. Le contrôle Sélecteur de champ possède les propriétés suivantes :

**Identificateur.** Identificateur unique du contrôle.

**Titre.** Titre facultatif qui apparaît au-dessus du contrôle. Pour les titres comportant plusieurs lignes, utilisez \n pour indiquer les sauts de ligne.

**Position du titre.** Indique la position du titre du contrôle. Les valeurs sont Top (Haut) et Left (Gauche) (Top est la valeur par défaut). Cette propriété s'applique uniquement lorsque le type de sélecteur est défini pour sélectionner un seul champ.

**Infobulle.** Texte d'infobulle facultatif qui apparaît lorsque l'utilisateur survole le contrôle. Le texte indiqué apparaît uniquement lorsque la souris survole la zone de titre du contrôle. Lorsque vous survolez les champs répertoriés, leur nom et leur libellé apparaissent.

**Touche mnémonique.** Caractère facultatif dans le titre à utiliser comme raccourci clavier pour activer le contrôle. Il est souligné dans le titre. Le raccourci est activé en appuyant sur Alt+[touche mnémonique].

**Type de sélecteur.** Indique si le sélecteur de champ dans la boîte de dialogue de noeud personnalisée peut être utilisé pour sélectionner un seul ou plusieurs champs dans la liste des champs.

**Type de séparateur.** Désigne le délimiteur entre les champs sélectionnés dans le script généré. Les séparateurs autorisés sont le blanc, la virgule et le signe plus (+). Vous pouvez aussi entrer un caractère unique arbitraire à utiliser comme séparateur.

**Minimum Fields (Nombre minimum de champs).** Nombre minimal de zones qui doivent être définies pour le contrôle, le cas échéant.

**Maximum Fields (Nombre maximum de champs).** Nombre maximal de zones qui peuvent être définies pour le contrôle, le cas échéant.

**Requis pour l'exécution.** Indique si une valeur est requise dans ce contrôle pour que l'exécution puisse continuer. Si **True** est spécifié, l'utilisateur de la boîte de dialogue de noeud doit indiquer une valeur pour le contrôle, sinon une erreur est générée lorsqu'il clique sur le bouton **OK**. Si **False** est spécifié, l'absence de valeur dans ce contrôle n'a aucune incidence sur l'état du bouton **OK**.

**Filtre de variable.** Permet de filtrer l'ensemble des champs affichés dans le contrôle. Vous pouvez filtrer en fonction du type et du niveau de mesure, et indiquer que des jeux de réponses multiples sont inclus dans la liste des champs. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue Filtre. Vous pouvez obtenir le même résultat en cliquant deux fois sur le contrôle Sélecteur de champ dans le canevas. Pour plus d'informations, voir «Filtrage des listes de champs».

**Source de champ.** Indique qu'un autre contrôle Sélecteur de champ est la source des champs du sélecteur de champ en cours. Lorsque la propriété Source de champ n'est pas définie, la source des champs est le jeu de données actif. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la source de champ.

**Script.** Indique le script généré et exécuté par ce contrôle au moment de l'exécution et pouvant être inséré dans le modèle de script.

- Vous pouvez spécifier tout script R ou Python for Spark valide. Pour les scripts multilignes ou les scripts longs, cliquez sur le bouton des points de suspension (...) et entrez votre script dans la boîte de dialogue Propriétés de script.
- La valeur `%ThisValue%` spécifie la valeur d'exécution du contrôle qui est la liste des champs. Il s'agit de la valeur par défaut.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle. La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.

**Définition de la source de la zone d'un sélecteur de champ :** La boîte de dialogue Source de la zone définit la source des zones affichées dans le sélecteur de champ. La source peut être un autre sélecteur de champ. Vous pouvez choisir d'afficher les zones qui sont dans les contrôles sélectionnés ou les zones du jeu de données actif qui ne sont pas dans le contrôle sélectionné.

## Filtrage des listes de champs

La boîte de dialogue Filtre, associée aux contrôles de sélecteur de champs, vous permet de filtrer les types de champs du jeu de données actif qui peuvent apparaître dans les listes. Vous pouvez également indiquer si des jeux de réponses multiples associés à l'ensemble de données actif sont inclus. Les champs incluent tous les formats numériques, à l'exception des formats de date et heure.

## Case à cocher

Un contrôle Case à cocher (Check Box) est une simple case à cocher qui peut générer et exécuter différents scripts R ou Python for Spark en fonction de l'état coché ou décoché de la case. Le contrôle Case à cocher possède les propriétés suivantes :

**Identificateur.** Identificateur unique du contrôle.

**Titre.** Titre facultatif qui apparaît au-dessus du contrôle. Pour les titres comportant plusieurs lignes, utilisez `\n` pour indiquer les sauts de ligne.

**Infobulle.** Texte d'infobulle facultatif qui apparaît lorsque l'utilisateur survole le contrôle.



**Touche mnémonique.** Caractère facultatif dans le titre à utiliser comme raccourci clavier pour activer le contrôle. Il est souligné dans le titre. Le raccourci est activé en appuyant sur Alt+[touche mnémonique].

**Valeur par défaut :** L'état par défaut de la case à cocher de contrôle, c'est-à-dire coché ou non coché.

**ScriptCoché/non coché.** Indique la le script R ou Python for Spark qui est généré et exécuté lorsque le contrôle est coché et lorsqu'il est non coché. Pour inclure le script dans le modèle de script, utilisez la valeur de la propriété Identificateur. La le script généré, à partir de la propriété Script ou Script vérifiée ou non vérifiée, est inséré aux emplacements spécifiés de l'identificateur. Par exemple, si l'identificateur est *checkbox1*, lors de l'exécution, les instances de *%checkbox1%* dans le modèle de script sont remplacées par la valeur de la propriété Script vérifié lorsque la case est cochée et par la valeur de la propriété Script non vérifié lorsque la case est non cochée.

- Vous pouvez spécifier tout script R ou Python for Spark valide. Pour les scripts multilignes ou les scripts longs, cliquez sur le bouton des points de suspension (...) et entrez votre script dans la boîte de dialogue Propriétés de script.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle. La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.

### Zone de liste déroulante

Le contrôle Zone de liste déroulante (Combo Box) permet de créer une liste déroulante pouvant générer et exécuter un script R ou Python for Spark propre à l'élément de liste sélectionné. Il est limité à une sélection unique. Le contrôle Zone de liste déroulante possède les propriétés suivantes :

**Identificateur.** Identificateur unique du contrôle. Identificateur à utiliser pour référencer le contrôle dans le modèle de script.

**Titre.** Titre facultatif qui apparaît au-dessus du contrôle. Pour les titres comportant plusieurs lignes, utilisez \n pour indiquer les sauts de ligne.

**Position du titre.** Indique la position du titre du contrôle. Les valeurs sont Top (Haut) et Left (Gauche) (Top est la valeur par défaut).

**Infobulle.** Texte d'infobulle facultatif qui apparaît lorsque l'utilisateur survole le contrôle.

**Éléments de liste.** Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue Propriétés des éléments de la liste, qui permet de spécifier les éléments de liste du contrôle. Vous pouvez aussi ouvrir la boîte de dialogue Propriétés des éléments de liste en double-cliquant sur le contrôle Zone de liste déroulante sur le canevas.

**Touche mnémonique.** Caractère facultatif dans le titre à utiliser comme raccourci clavier pour activer le contrôle. Il est souligné dans le titre. Le raccourci est activé en appuyant sur Alt+[touche mnémonique].

**Modifiable.** Indique si le contrôle Zone de liste déroulante est modifiable. Lorsque le contrôle est modifiable, une valeur personnalisé peut être entrée au moment de l'exécution.

**Script.**Spécifie le script R ou Python for Spark pouvant être généré par ce contrôle lors de l'exécution et inséré dans le modèle de script.

- La valeur *%ThisValue%* spécifie la valeur d'exécution du contrôle. Elle est utilisée par défaut. Si les éléments de liste sont définis manuellement, la valeur d'exécution est la valeur de la propriété Script de l'élément de liste sélectionné. Si les éléments de liste sont basés sur un contrôle de liste cible, la valeur d'exécution est la valeur de l'élément de liste sélectionné. Pour des contrôles de zone de liste à

sélection multiple, la valeur d'exécution est une liste séparée par des espaces des éléments sélectionnés. Pour plus d'informations, voir «Spécification d'éléments de liste pour les zones de liste déroulante et les zones de liste».

- Vous pouvez spécifier tout script R ou Python for Spark valide. Pour les scripts multilignes ou les scripts longs, cliquez sur le bouton des points de suspension (...) et entrez votre script dans la boîte de dialogue Propriétés de script.

**Traitement des guillemets :** Définit la gestion des guillemets dans la valeur d'exécution de `%%ThisValue%%` lorsque la propriété Script contient `%%ThisValue%%` dans la chaîne entre guillemets. Dans ce contexte, une chaîne entre guillemets est une chaîne située entre apostrophes ou doubles apostrophes. La gestion des guillemets s'applique uniquement aux apostrophes qui sont identiques aux apostrophes qui entourent `%%ThisValue%%`. Les types de gestion de guillemets disponibles sont les suivants :

#### Python

Les apostrophes présentes dans la valeur d'exécution de `%%ThisValue%%` qui sont similaires aux apostrophes fermantes sont associées au caractère d'échappement barre oblique inversée (\). Par exemple, si la propriété Script est `'%%ThisValue%%'` et que la valeur d'exécution de la zone de liste déroulante est Valeur d'apostrophe, le script généré est `'Valeur d\'apostrophe'`. Notez qu'il n'y a pas de gestion des apostrophes si `%%ThisValue%%` est placé entre des apostrophes triplées.

#### R

Les apostrophes présentes dans la valeur d'exécution de `%%ThisValue%%` qui sont similaires aux apostrophes fermantes sont associées au caractère d'échappement barre oblique inversée (\). Par exemple, si la propriété Script est `'%%ThisValue%%'` et que la valeur d'exécution de la zone de liste déroulante est Valeur d'apostrophe, le script généré est `'Valeur d\'apostrophe'`.

#### Aucun

Les apostrophes présentes dans la valeur d'exécution de `%%ThisValue%%` qui sont similaires aux apostrophes fermantes sont conservées sans aucune modification.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle. La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.

**Spécification d'éléments de liste pour les zones de liste déroulante et les zones de liste :** La boîte de dialogue Propriétés des éléments de la liste permet de spécifier les éléments de liste d'un contrôle de zone de liste déroulante ou de zone de liste.

**Valeurs définies manuellement.** Permet de spécifier explicitement chacun des éléments de liste.

- **Identificateur.** Identificateur unique de l'élément de liste.
- **Nom.** Nom qui apparaît dans la liste pour cet élément. Le nom est un champ obligatoire.
- **Défaut.** Pour une zone de liste déroulante, indique si l'élément de liste est l'élément par défaut affiché dans la zone de liste déroulante. Pour une zone de liste, indique si l'élément de liste est sélectionné par défaut.
- **Script.** Spécifie le script R ou Python for Spark généré lorsque la liste est sélectionnée.
- Vous pouvez spécifier tout script R ou Python for Spark valide. Pour les scripts multilignes ou les scripts longs, cliquez sur le bouton des points de suspension (...) et entrez votre script dans la boîte de dialogue Propriétés de script.

**Remarque :** Vous pouvez ajouter un nouvel élément de liste dans la ligne vierge en dessous de la liste existante. La saisie de propriétés autres que Identificateur génère un identificateur unique que vous pouvez conserver ou modifier. Pour supprimer un élément de liste, cliquez sur la cellule *Identificateur* correspondant à l'élément et sélectionnez Supprimer.

## Zone de liste

Le contrôle Zone de liste permet d'afficher une liste des éléments qui prennent en charge la sélection unique ou multiple et de générer le script R ou Python for Spark propre à l'élément ou aux éléments sélectionnés. Le contrôle Zone de liste possède les propriétés suivantes :

**Identificateur.** Identificateur unique du contrôle. Identificateur à utiliser pour référencer le contrôle dans le modèle de script.

**Titre.** Titre facultatif qui apparaît au-dessus du contrôle. Pour les titres comportant plusieurs lignes, utilisez \n pour indiquer les sauts de ligne.

**Infobulle.** Texte d'infobulle facultatif qui apparaît lorsque l'utilisateur survole le contrôle.

**Éléments de liste.** Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue Propriétés des éléments de la liste, qui permet de spécifier les éléments de liste du contrôle. Vous pouvez aussi ouvrir la boîte de dialogue Propriétés des éléments de liste en double-cliquant sur le contrôle Zone de liste sur le canevas.

**Touche mnémonique.** Caractère facultatif dans le titre à utiliser comme raccourci clavier pour activer le contrôle. Il est souligné dans le titre. Le raccourci est activé en appuyant sur Alt+[touche mnémonique].

**Type de zone de liste :** Indique si la zone de liste prend en charge la sélection unique ou la sélection multiple. Vous pouvez également indiquer que les éléments s'affichent sous la forme d'une liste de cases à cocher.

**Type de séparateur :** Définit le délimiteur des éléments de liste sélectionnés dans le script généré. Les séparateurs admis sont l'espace, la virgule et le signe plus (+). Vous pouvez également définir comme séparateur un caractère unique arbitraire.

**Minimum sélectionné :** Nombre minimal d'éléments qui doivent être sélectionnés dans le contrôle, le cas échéant.

**Maximum sélectionné :** Nombre maximal d'éléments qui peuvent être sélectionnés dans le contrôle, le cas échéant.

**Script.**Spécifie le script R ou Python for Spark pouvant être généré par ce contrôle lors de l'exécution et inséré dans le modèle de script.

- La valeur `%%ThisValue%%` spécifie la valeur d'exécution du contrôle. Elle est utilisée par défaut. Si les éléments de liste sont définis manuellement, la valeur d'exécution est la valeur de la propriété Script de l'élément de liste sélectionné. Si les éléments de liste sont basés sur un contrôle de liste cible, la valeur d'exécution est la valeur de l'élément de liste sélectionné. Pour les contrôles de zone de liste à sélection multiple, la valeur au moment de l'exécution est la liste des éléments sélectionnés, séparés par le type de séparateur défini (l'espace est la valeur par défaut). Pour plus d'informations, voir «Spécification d'éléments de liste pour les zones de liste déroulante et les zones de liste», à la page 36.
- Vous pouvez spécifier tout script R ou Python for Spark valide. Pour les scripts multilignes ou les scripts longs, cliquez sur le bouton des points de suspension (...) et entrez votre script dans la boîte de dialogue Propriétés de script.

**Traitement des guillemets :** Définit la gestion des guillemets dans la valeur d'exécution de `%%ThisValue%%` lorsque la propriété Script contient `%%ThisValue%%` dans la chaîne entre guillemets. Dans ce contexte, une chaîne entre guillemets est une chaîne située entre apostrophes ou doubles apostrophes. La gestion des guillemets s'applique uniquement aux apostrophes qui sont identiques aux apostrophes qui entourent `%%ThisValue%%`. Les types de gestion de guillemets disponibles sont les suivants :

### Python

Les apostrophes présentes dans la valeur d'exécution de %%ThisValue%% qui sont similaires aux apostrophes fermantes sont associées au caractère d'échappement barre oblique inversée (\). Par exemple, si la propriété Script est '%%ThisValue%%' et si l'élément de liste sélectionné est Valeur d'exécution, le script généré est 'Valeur d\'exécution'. Notez qu'il n'y a pas de gestion des apostrophes si %%ThisValue%% est placé entre des apostrophes triplées.

### R

Les apostrophes présentes dans la valeur d'exécution de %%ThisValue%% qui sont similaires aux apostrophes fermantes sont associées au caractère d'échappement barre oblique inversée (\). Par exemple, si la propriété Script est '%%ThisValue%%' et si l'élément de liste sélectionné est Valeur d'exécution, le script généré est 'Valeur d\'exécution'.

### Aucun

Les apostrophes présentes dans la valeur d'exécution de %%ThisValue%% qui sont similaires aux apostrophes fermantes sont conservées sans aucune modification.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle. La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.

### Contrôle du texte

Le contrôle de texte est une simple zone de texte pouvant accepter des entrées arbitraires et dont les propriétés sont les suivantes :

**Identificateur.** Identificateur unique du contrôle. Identificateur à utiliser pour référencer le contrôle dans le modèle de script.

**Titre.** Titre facultatif qui apparaît au-dessus du contrôle. Pour les titres comportant plusieurs lignes, utilisez \n pour indiquer les sauts de ligne.

**Position du titre.** Indique la position du titre du contrôle. Les valeurs sont Top (Haut) et Left (Gauche) (Top est la valeur par défaut).

**Infobulle.** Texte d'infobulle facultatif qui apparaît lorsque l'utilisateur survole le contrôle.

**Touche mnémotechnique.** Caractère facultatif dans le titre à utiliser comme raccourci clavier pour activer le contrôle. Il est souligné dans le titre. Le raccourci est activé en appuyant sur Alt+[touche mnémotechnique].

**Contenu de texte.** Indique si les contenus sont arbitraires ou si la zone de texte doit contenir une chaîne conforme aux règles des noms de zone d'IBM SPSS Modeler.

**Valeur par défaut :** Contenu par défaut de la zone de texte.

**Largeur.** Indique la largeur de la zone de texte du contrôle en caractères. Les valeurs admises sont des entiers positifs. Une valeur vide signifie que la largeur est déterminée automatiquement.

**Requis pour l'exécution.** Indique si une valeur est requise dans ce contrôle pour que l'exécution puisse continuer. Si **True** est spécifié, l'utilisateur de la boîte de dialogue de noeud doit indiquer une valeur pour le contrôle, sinon une erreur est générée lorsqu'il clique sur le bouton **OK**. Si **False** est spécifié, l'absence de valeur dans ce contrôle n'a aucune incidence sur l'état du bouton **OK**. La valeur par défaut est **False**.

**Script.** Spécifie le script R ou Python for Spark devant être généré et exécuté par ce contrôle lors de l'exécution et pouvant être inséré dans le modèle de script.

- Vous pouvez spécifier tout script R ou Python for Spark valide. Pour les scripts multilignes ou les scripts longs, cliquez sur le bouton des points de suspension (...) et entrez votre script dans la boîte de dialogue Propriétés de script.
- La valeur `%%ThisValue%%` indique la valeur d'exécution du contrôle, qui est le contenu de la zone de texte. Il s'agit de la valeur par défaut.
- Si la propriété Script inclut `%%ThisValue%%` et que la valeur d'exécution de la zone de texte est vide, le contrôle de zone de texte ne génère pas de script.

**Traitement des guillemets :** Définit la gestion des guillemets dans la valeur d'exécution de `%%ThisValue%%` lorsque la propriété Script contient `%%ThisValue%%` dans la chaîne entre guillemets. Dans ce contexte, une chaîne entre guillemets est une chaîne située entre apostrophes ou doubles apostrophes. La gestion des guillemets s'applique uniquement aux apostrophes qui sont identiques aux apostrophes qui entourent `%%ThisValue%%`. Les types de gestion de guillemets disponibles sont les suivants :

#### **Python**

Les apostrophes présentes dans la valeur d'exécution de `%%ThisValue%%` qui sont similaires aux apostrophes fermantes sont associées au caractère d'échappement barre oblique inversée (`\`). Par exemple, si la propriété Script est `'%%ThisValue%%'` et que la valeur d'exécution du contrôle de texte est Valeur d'apostrophe, le script généré est `'Valeur d\'apostrophe'`. Il n'y a pas de gestion des apostrophes si `%%ThisValue%%` est placé entre des apostrophes triplées.

#### **R**

Les apostrophes présentes dans la valeur d'exécution de `%%ThisValue%%` qui sont similaires aux apostrophes fermantes sont associées au caractère d'échappement barre oblique inversée (`\`). Par exemple, si la propriété Script est `'%%ThisValue%%'` et que la valeur d'exécution du contrôle de texte est Valeur d'apostrophe, le script généré est `'Valeur d\'apostrophe'`.

#### **Aucun**

Les apostrophes présentes dans la valeur d'exécution de `%%ThisValue%%` qui sont similaires aux apostrophes fermantes sont conservées sans aucune modification.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle. La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.

### **Contrôle des nombres**

Contrôle des nombres est une zone de texte permettant la saisie d'une valeur numérique et qui possède les propriétés suivantes :

**Identificateur.** Identificateur unique du contrôle. Identificateur à utiliser pour référencer le contrôle dans le modèle de script.

**Titre.** Titre facultatif qui apparaît au-dessus du contrôle. Pour les titres comportant plusieurs lignes, utilisez `\n` pour indiquer les sauts de ligne.

**Position du titre.** Indique la position du titre du contrôle. Les valeurs sont Top (Haut) et Left (Gauche) (Top est la valeur par défaut).

**Infobulle.** Texte d'infobulle facultatif qui apparaît lorsque l'utilisateur survole le contrôle.

**Touche mnémonique.** Caractère facultatif dans le titre à utiliser comme raccourci clavier pour activer le contrôle. Il est souligné dans le titre. Le raccourci est activé en appuyant sur `Alt+[touche mnémonique]`.

**Type numérique.** Indique les restrictions éventuelles imposées lors de la saisie. La valeur Réel indique qu'aucune restriction n'est appliquée aux valeurs saisies, en dehors du fait qu'elles soient numériques. La valeur Entier indique que la valeur doit être un entier.

**Entrée spinner.** Indique si le contrôle est affiché sous forme de spinner (bouton fléché). La valeur par défaut est False.

**Incrément.** Incrément utilisé lorsque le contrôle est affiché sous forme de spinner.

**Valeur par défaut :** Valeur par défaut, le cas échéant.

**Valeur minimale.** Valeur minimale autorisée, le cas échéant.

**Valeur maximale.** Valeur maximale autorisée, le cas échéant.

**Largeur.** Indique la largeur de la zone de texte du contrôle en caractères. Les valeurs admises sont des entiers positifs. Une valeur vide signifie que la largeur est déterminée automatiquement.

**Requis pour l'exécution.** Indique si une valeur est requise dans ce contrôle pour que l'exécution puisse continuer. Si **True** est spécifié, l'utilisateur de la boîte de dialogue de noeud doit indiquer une valeur pour le contrôle, sinon une erreur est générée lorsqu'il clique sur le bouton **OK**. Si **False** est spécifié, l'absence de valeur dans ce contrôle n'a aucune incidence sur l'état du bouton **OK**. La valeur par défaut est **False**.

**Script.** Spécifie le script R ou Python for Spark devant être généré et exécuté par ce contrôle lors de l'exécution et pouvant être inséré dans le modèle de script.

- Vous pouvez spécifier tout script R ou Python for Spark valide. Pour les scripts multilignes ou les scripts longs, cliquez sur le bouton des points de suspension (...) et entrez votre script dans la boîte de dialogue Propriétés de script.
- La valeur `%%ThisValue%%` indique la valeur d'exécution du contrôle, qui est une valeur numérique. Il s'agit de la valeur par défaut.
- Si la propriété Script inclut `%%ThisValue%%` et que la valeur d'exécution du contrôle des nombres est vide, le contrôle des nombres ne génère pas de script.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle. La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.

## Date

La contrôle Date est un contrôle spinner permettant de spécifier les valeurs de date/heure (date, heure, horodatage). Le contrôle Date possède les propriétés suivantes :

**Identificateur.** Identificateur unique du contrôle. Identificateur à utiliser pour référencer le contrôle dans le modèle de script.

**Titre.** Titre facultatif qui apparaît au-dessus du contrôle. Pour les titres comportant plusieurs lignes, utilisez `\n` pour indiquer les sauts de ligne.

**Position du titre.** Indique la position du titre du contrôle. Les valeurs sont Top (Haut) et Left (Gauche) (Top est la valeur par défaut).

**Infobulle.** Texte d'infobulle facultatif qui apparaît lorsque l'utilisateur survole le contrôle.

**Touche mnémonique.** Caractère facultatif dans le titre à utiliser comme raccourci clavier pour activer le contrôle. Il est souligné dans le titre. Le raccourci est activé en appuyant sur `Alt+[touche mnémonique]`.

**Type.** Indique si le contrôle est pour les dates, les heures ou les horodatages.

**Date** Le contrôle indique une date calendaire au format aaaa-mm-jj. La valeur d'exécution par défaut est spécifiée par la propriété Valeur par défaut.

**Heure** Le contrôle indique l'heure au format hh:mm:ss. La valeur d'exécution par défaut est l'heure actuelle.

**Date/heure**

Le contrôle indique une date et une heure au format aaaa-mm-jj hh:mm:ss. La valeur d'exécution par défaut est la date et l'heure actuelles.

**Valeur par défaut :** Valeur d'exécution par défaut du contrôle lorsque le type est Date. Vous pouvez paramétrer l'affichage de la date actuelle ou d'une date particulière.

**Script.** Spécifie le script R ou Python for Spark devant être généré et exécuté par ce contrôle lors de l'exécution et pouvant être inséré dans le modèle de script.

- Vous pouvez spécifier tout script R ou Python for Spark valide. Pour les scripts multilignes ou les scripts longs, cliquez sur le bouton des points de suspension (...) et entrez votre script dans la boîte de dialogue Propriétés de script.
- La valeur `%ThisValue%` spécifie la valeur d'exécution du contrôle. Il s'agit de la valeur par défaut.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle. La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.

**Remarque :** Le contrôle Date n'est pas pris en charge dans les versions d'IBM SPSS Modeler antérieures à la version 18.

### Texte sécurisé

Le contrôle Texte sécurisé est une zone de texte qui masque l'entrée utilisateur avec des astérisques.

**Identificateur.** Identificateur unique du contrôle. Identificateur à utiliser pour référencer le contrôle dans le modèle de script.

**Titre.** Titre facultatif qui apparaît au-dessus du contrôle. Pour les titres comportant plusieurs lignes, utilisez `\n` pour indiquer les sauts de ligne.

**Position du titre.** Indique la position du titre du contrôle. Les valeurs sont Top (Haut) et Left (Gauche) (Top est la valeur par défaut).

**Infobulle.** Texte d'infobulle facultatif qui apparaît lorsque l'utilisateur survole le contrôle.

**Touche mnémonique.** Caractère facultatif dans le titre à utiliser comme raccourci clavier pour activer le contrôle. Il est souligné dans le titre. Le raccourci est activé en appuyant sur `Alt+[touche mnémonique]`.

**Largeur.** Indique la largeur de la zone de texte du contrôle en caractères. Les valeurs admises sont des entiers positifs. Une valeur vide signifie que la largeur est déterminée automatiquement.

**Requis pour l'exécution.** Indique si une valeur est requise dans ce contrôle pour que l'exécution puisse continuer. Si **True** est spécifié, l'utilisateur de la boîte de dialogue de noeud doit indiquer une valeur pour le contrôle, sinon une erreur est générée lorsqu'il clique sur le bouton **OK**. Si **False** est spécifié, l'absence de valeur dans ce contrôle n'a aucune incidence sur l'état du bouton **OK**. La valeur par défaut est **False**.

**Script.** Spécifie le script R ou Python for Spark devant être généré et exécuté par ce contrôle lors de l'exécution et pouvant être inséré dans le modèle de script.

- Vous pouvez spécifier tout script R ou Python for Spark valide. Pour les scripts multilignes ou les scripts longs, cliquez sur le bouton des points de suspension (...) et entrez votre script dans la boîte de dialogue Propriétés de script.
- La valeur `%%ThisValue%%` indique la valeur d'exécution du contrôle, qui est le contenu de la zone de texte. Il s'agit de la valeur par défaut.
- Si la propriété Script inclut `%%ThisValue%%` et que la valeur d'exécution du contrôle de texte sécurisé est vide, le contrôle de texte sécurisé ne génère pas de script R ou Python for Spark.

**Traitement des guillemets :** Définit la gestion des guillemets dans la valeur d'exécution de `%%ThisValue%%` lorsque la propriété Script contient `%%ThisValue%%` dans la chaîne entre guillemets. Dans ce contexte, une chaîne entre guillemets est une chaîne située entre apostrophes ou doubles apostrophes. La gestion des guillemets s'applique uniquement aux apostrophes qui sont identiques aux apostrophes qui entourent `%%ThisValue%%` et uniquement lorsque Encrypt passed value=False. Les types de gestion de guillemets disponibles sont les suivants :

#### **Python**

Les apostrophes présentes dans la valeur d'exécution de `%%ThisValue%%` qui sont similaires aux apostrophes fermantes sont associées au caractère d'échappement barre oblique inversée (\). Par exemple, si la propriété Script est '`%%ThisValue%%`' et que la valeur d'exécution du contrôle est Valeur d'apostrophe, le script généré est 'Valeur d\'apostrophe'. Il n'y a pas de gestion des apostrophes si `%%ThisValue%%` est placé entre des apostrophes triplées.

#### **R**

Les apostrophes présentes dans la valeur d'exécution de `%%ThisValue%%` qui sont similaires aux apostrophes fermantes sont associées au caractère d'échappement barre oblique inversée (\). Par exemple, si la propriété Script est '`%%ThisValue%%`' et que la valeur d'exécution du contrôle est Valeur d'apostrophe, le script généré est 'Valeur d\'apostrophe'.

#### **Aucun**

Les apostrophes présentes dans la valeur d'exécution de `%%ThisValue%%` qui sont similaires aux apostrophes fermantes sont conservées sans aucune modification.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle. La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.

**Remarque :** Le contrôle Texte sécurisé n'est pas pris en charge dans les versions d'IBM SPSS Modeler antérieures à la version 18.

### **Contrôle de texte statique**

Le contrôle Texte statique permet d'ajouter un bloc de texte à la boîte de dialogue de noeud et il possède les propriétés suivantes :

**Identificateur.** Identificateur unique du contrôle.

**Titre.** Contenu du bloc de texte. Pour le contenu à plusieurs lignes, utilisez \n pour spécifier un retour à la ligne.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle.



La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.

## Sélecteur de couleur

Le contrôle Sélecteur de couleur (Color Picker) est une interface utilisateur permettant de spécifier une couleur et de générer la valeur RVB associée. Le contrôle Sélecteur de couleur possède les propriétés suivantes :

**Identificateur.** Identificateur unique du contrôle. Identificateur à utiliser pour référencer le contrôle dans le modèle de script.

**Titre.** Titre facultatif qui apparaît au-dessus du contrôle. Pour les titres comportant plusieurs lignes, utilisez \n pour indiquer les sauts de ligne.

**Position du titre.** Indique la position du titre du contrôle. Les valeurs sont Top (Haut) et Left (Gauche) (Top est la valeur par défaut).

**Infobulle.** Texte d'infobulle facultatif qui apparaît lorsque l'utilisateur survole le contrôle.

**Touche mnémonique.** Caractère facultatif dans le titre à utiliser comme raccourci clavier pour activer le contrôle. Il est souligné dans le titre. Le raccourci est activé en appuyant sur Alt+[touche mnémonique].

**Script.** Spécifie le script R ou Python for Spark devant être généré et exécuté par ce contrôle lors de l'exécution et pouvant être inséré dans le modèle de script.

- Vous pouvez spécifier tout script R ou Python for Spark valide. Pour les scripts multilignes ou les scripts longs, cliquez sur le bouton des points de suspension (...) et entrez votre script dans la boîte de dialogue Propriétés de script.
- La valeur `%%ThisValue%%` indique la valeur d'exécution du contrôle, qui est la valeur RVB de la couleur sélectionnée. La valeur RVB est représentée sous la forme d'une liste d'entiers séparés par des espaces dans l'ordre suivant : valeur R, valeur V, valeur B.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle. La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.

**Remarque :** Le contrôle Sélecteur de couleur n'est pas pris en charge dans les versions d'IBM SPSS Modeler antérieures à la version 18.

## Table

Le contrôle Table crée une table avec un nombre fixe de colonnes et un nombre variable de lignes ajoutées au moment de l'exécution. Le contrôle Table possède les propriétés suivantes :

**Identificateur.** Identificateur unique du contrôle. Identificateur à utiliser pour référencer le contrôle dans le modèle de script.

**Titre.** Titre facultatif qui apparaît au-dessus du contrôle. Pour les titres comportant plusieurs lignes, utilisez \n pour indiquer les sauts de ligne.

**Infobulle.** Texte d'infobulle facultatif qui apparaît lorsque l'utilisateur survole le contrôle.

**Touche mnémonique.** Caractère facultatif dans le titre à utiliser comme raccourci clavier pour activer le contrôle. Il est souligné dans le titre. Le raccourci est activé en appuyant sur Alt+[touche mnémonique].

**Boutons de réorganisation (Reorder Buttons).** Indique si des boutons de déplacement vers le haut et vers le bas sont ajoutés à la table. Ces boutons sont utilisés lors de l'exécution pour réorganiser les lignes de la table.

**Colonnes de table (Table Columns).** Cliquez sur le bouton de points de suspension (...) pour ouvrir la boîte de dialogue Colonnes de table dans laquelle vous spécifiez les colonnes de la table.

**Minimum Rows (Nombre maximum de lignes).** Nombre minimum de lignes dans la table.

**Maximum Rows (Nombre maximum de lignes).** Nombre maximum de lignes dans la table.

**Requis pour l'exécution.** Indique si une valeur est requise dans ce contrôle pour que l'exécution puisse continuer. Si **True** est spécifié, l'utilisateur de la boîte de dialogue de noeud doit indiquer une valeur pour le contrôle, sinon une erreur est générée lorsqu'il clique sur le bouton **OK**. Si **False** est spécifié, l'absence de valeur dans ce contrôle n'a aucune incidence sur l'état du bouton **OK**.

**Script.**Spécifie le script R ou Python for Spark pouvant être généré par ce contrôle lors de l'exécution et inséré dans le modèle de script.

- La valeur `%%ThisValue%%` spécifie la valeur d'exécution du contrôle. Elle est utilisée par défaut. La valeur d'exécution est une liste séparée par des espaces des scripts générés par chaque colonne de la table, en commençant par la colonne la plus à gauche. Si la propriété Script inclut `%%ThisValue%%` et qu'aucune des colonnes ne génère de script, la table dans son ensemble ne génère aucun script.
- Vous pouvez spécifier tout script R ou Python for Spark valide. Pour les scripts multilignes ou les scripts longs, cliquez sur le bouton des points de suspension (...) et entrez votre script dans la boîte de dialogue Propriétés de script.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle. La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.

**Remarque :** Le contrôle Table n'est pas pris en charge dans les versions d'IBM SPSS Modeler antérieures à la version 18.

**Définition de colonnes pour les contrôles de tableau :** La boîte de dialogue Colonnes du tableau définit les propriétés des colonnes du contrôle de tableau.

**Identificateur.** Identificateur unique de la colonne.

**Nom de colonne.** Nom de la colonne tel qu'il figure dans le tableau.

**Contenu.** Type des données de la colonne. La valeur **Réel** spécifie qu'il n'y a aucune restriction sur la valeur saisie, si ce n'est qu'elle doit être une valeur numérique. La valeur **Entier** spécifie que la valeur doit être un entier. La valeur **Tout** spécifie qu'il n'y a aucune restriction sur la valeur saisie. La valeur **Nom de variable** indique que la valeur doit respecter les critères de validité des noms de variable d'IBM SPSS Statistics.

**Valeur par défaut :** Valeur par défaut de la colonne, le cas échéant, lors de l'ajout de nouvelles lignes dans le tableau en phase d'exécution.

**Type de séparateur.** Définit le délimiteur des valeurs de la colonne dans le script généré. Les séparateurs admis sont l'espace, la virgule et le signe plus (+). Vous pouvez également définir comme séparateur un caractère unique arbitraire.

**Entre guillemets.** Indique si chaque valeur de la colonne est mis entre guillemets dans le script généré.

**Traitement des guillemets :** Définit le traitement des guillemets dans les entrées de cellule de la colonne lorsque la propriété Entre guillemets a la valeur true. Le traitement des guillemets s'applique uniquement aux guillemets des valeurs des cellules. Les types de gestion de guillemets disponibles sont les suivants :

**Python**

Les guillemets des valeurs des cellules sont précédés d'un caractère d'échappement qui est une barre oblique inversée (\). Par exemple, si la valeur d'une cellule est Cette valeur "entre guillemets", le script généré est "Cette valeur \"entre guillemets\"".

**R**

Les guillemets des valeurs des cellules sont précédés d'un caractère d'échappement qui est une barre oblique inversée (\). Par exemple, si la valeur d'une cellule est Cette valeur "entre guillemets", le script généré est "Cette valeur \"entre guillemets\"".

**Aucun**

Les guillemets des valeurs des cellules sont conservés tels quels.

**Largeur(car.).** Définit la largeur de la colonne, en caractères. Les valeurs autorisées sont des entiers non négatifs.

**Script.** Spécifie le script R ou le script Python for Spark généré par cette colonne en phase d'exécution. Le script généré pour l'ensemble du tableau est une liste dont les éléments, séparés par des espaces, sont les scripts générés par chaque colonne du tableau, en commençant par la colonne la plus à gauche.

- Vous pouvez spécifier tout script R ou Python for Spark valide. Pour les scripts multilignes ou les scripts longs, cliquez sur le bouton des points de suspension (...) et entrez votre script dans la boîte de dialogue Propriétés de script.
- La valeur `%ThisValue%` spécifie la valeur d'exécution de la colonne, qui est la liste de valeurs de la colonne, séparées par le délimiteur choisi.
- Si la propriété Script de la colonne inclut `%ThisValue%` et si la valeur d'exécution de la colonne est vide, la colonne ne génère pas de script.

**Remarque :** Vous pouvez ajouter une ligne pour une nouvelle colonne de tableau dans la ligne vierge située sous la liste existante dans la boîte de dialogue Colonnes du tableau. La saisie de propriétés différentes de l'identificateur génère un identificateur unique que vous pouvez conserver ou modifier. Vous pouvez supprimer une colonne de tableau en cliquant sur la cellule d'identificateur de la colonne et en appuyant sur Supprimer.

**Lien à un contrôle**

Vous pouvez lier un contrôle Table à un contrôle Sélecteur de champ. Lorsqu'un contrôle Table est lié à un contrôle Sélecteur de champ, il y a une ligne dans la table pour chaque champ du sélecteur de champ. Les lignes sont ajoutées à la table en ajoutant des champs au sélecteur de champ. Les lignes sont supprimées de la table en supprimant les champs du sélecteur de champ. Un contrôle Table lié peut être utilisé, par exemple, pour spécifier les propriétés des champs sélectionnés dans le sélecteur de champ.

Pour activer les liaisons, la table doit contenir une colonne avec Nom de variable pour la propriété Contenu et le canevas doit comporter au moins un contrôle Sélecteur de champ.

Pour associer un contrôle Table à un sélecteur de champ, spécifiez ce dernier dans la liste des contrôles disponibles du groupe Lien à un contrôle dans la boîte de dialogue Colonnes de table. Sélectionnez ensuite la colonne, appelée **Colonne liée**, qui définit le lien. Lorsque la table est présentée, la colonne liée affiche les champs actuels dans le sélecteur de champ. Seuls les sélecteurs de champ à zones multiples peuvent lui être liés.

## Groupe d'éléments

Le contrôle Groupe d'éléments est un conteneur d'autres contrôles qui permet de grouper et de contrôler le script généré à partir de plusieurs contrôles. Par exemple, vous pouvez disposer d'un ensemble de cases à cocher qui spécifient des paramètres facultatifs pour une sous-commande, et souhaiter générer le script de la sous-commande si une case au moins est cochée. Pour cela, utilisez un contrôle Groupe d'éléments comme conteneur pour les contrôles de case à cocher. Un groupe d'éléments peut contenir les types de contrôles suivants : sélecteur de champ, case à cocher, zone de liste déroulante, zone de liste, contrôle de texte, contrôle de nombre, texte statique, groupe de boutons d'option et navigateur de fichiers. Le contrôle Groupe d'éléments possède les propriétés suivantes :

**Identificateur.** Identificateur unique du contrôle. Identificateur à utiliser pour référencer le contrôle dans le modèle de script.

**Titre.** Titre facultatif pour le groupe. Pour les titres comportant plusieurs lignes, utilisez \n pour indiquer les sauts de ligne.

**Script.** Spécifie le script R ou Python for Spark devant être généré et exécuté par ce contrôle lors de l'exécution et pouvant être inséré dans le modèle de script.

- Vous pouvez spécifier tout script R ou Python for Spark valide. Pour les scripts multilignes ou les scripts longs, cliquez sur le bouton des points de suspension (...) et entrez votre script dans la boîte de dialogue Propriétés de script.
- Vous pouvez inclure des identificateurs pour les contrôles contenus dans le groupe d'éléments. Lors de l'exécution, les identificateurs sont remplacés par le script R ou Python for Spark généré par les contrôles.
- La valeur `%%ThisValue%%` génère une liste séparée par des espaces du script R ou Python généré par chaque contrôle dans le groupe d'éléments, dans l'ordre d'apparition dans le groupe (du haut vers le bas). Il s'agit de la valeur par défaut. Si la propriété Script inclut `%%ThisValue%%` et qu'aucun script n'est généré par les contrôles dans le groupe d'éléments, ce dernier ne génère pas de script.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle. La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.

## Groupe de boutons radio

Le contrôle Groupe de boutons radio est un conteneur d'ensemble de boutons radios pouvant contenir chacun un ensemble de contrôles imbriqués. Il possède les propriétés suivantes :

**Identificateur.** Identificateur unique du contrôle. Identificateur à utiliser pour référencer le contrôle dans le modèle de script.

**Titre.** Titre facultatif pour le groupe. Pour les titres comportant plusieurs lignes, utilisez \n pour indiquer les sauts de ligne.

**Infobulle.** Texte d'infobulle facultatif qui apparaît lorsque l'utilisateur survole le contrôle.

**Boutons radio.** Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue Propriétés du groupe de boutons radio, qui permet de spécifier les propriétés des boutons radio, d'ajouter des boutons au groupe ou d'en supprimer. La capacité d'imbriquer des contrôles sous un bouton radio donné est une propriété du bouton radio qui est définie dans la boîte de dialogue Propriétés du groupe de boutons radio. Vous pouvez également ouvrir la boîte de dialogue Propriétés du groupe de boutons radio en cliquant deux fois sur le contrôle Groupe de boutons radio dans le canevas.

**Script.** Spécifie le script R ou Python for Spark généré par ce contrôle lors de l'exécution et pouvant être inséré dans le modèle de script.

- Vous pouvez spécifier tout script R ou Python for Spark valide. Pour les scripts multilignes ou les scripts longs, cliquez sur le bouton des points de suspension (...) et entrez votre script dans la boîte de dialogue Propriétés de script.
- La valeur `%%ThisValue%%` spécifie la valeur d'exécution du groupe de boutons radio qui est la valeur de la propriété Script du bouton radio sélectionné. Il s'agit de la valeur par défaut. Si la propriété Script R inclut `%%ThisValue%%` et qu'aucun script n'est généré par le bouton radio sélectionné, le groupe de boutons radio ne génère pas de script.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle. La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.

**Définition des boutons radio :** La boîte de dialogue Propriétés des groupes de boutons radio permet de spécifier un groupe de boutons radio.

**Identificateur.** Identificateur unique du bouton radio.

**Nom de colonne.** Nom qui apparaît en regard du bouton radio. Le nom est un champ obligatoire.

**Infobulle.** Texte d'infobulle facultatif qui apparaît lorsque l'utilisateur survole le contrôle.

**Touche mnémonique.** Un caractère optionnel du nom à utiliser en tant que mnémonique. Le caractère spécifié doit exister dans le nom.

**Groupe imbriqué.** Indique si d'autres contrôles peuvent être imbriqués sous ce bouton radio. La valeur par défaut est false. Lorsque la propriété Groupe imbriqué est définie sur true, une zone d'insertion rectangulaire s'affiche, imbriquée et mise en retrait, sous le bouton radio associé. Les contrôles suivants peuvent être imbriqués sous un bouton radio : sélecteur de champ, case à cocher, contrôle de texte, texte statique, contrôle de nombre, zone de liste déroulante, zone de liste et navigateur de fichiers.

**Défaut.** Indique si le bouton radio est la sélection par défaut.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle. La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.

**Script.** Spécifie le script R ou Python for Spark généré lorsque le bouton radio est sélectionné.

- Vous pouvez spécifier tout script R ou Python for Spark valide. Pour les scripts multilignes ou les scripts longs, cliquez sur le bouton des points de suspension (...) et entrez votre script dans la boîte de dialogue Propriétés de script.
- Pour les boutons radio contenant des contrôles imbriqués, la valeur `%%ThisValue%%` génère une liste séparée par des espaces du script R ou Python for Spark généré par chaque contrôle imbriqué, dans l'ordre d'apparition sous le bouton radio (du haut vers le bas).

Vous pouvez ajouter un nouveau bouton radio dans la ligne vide au bas de la liste existante. La saisie de propriétés autres que Identificateur génère un identificateur unique que vous pouvez conserver ou modifier. Pour supprimer un bouton radio, cliquez sur la cellule *Identificateur* correspondant au bouton radio et sélectionnez Supprimer.

## Groupes de cases à cocher

Le contrôle Groupe de cases à cocher est le conteneur d'un ensemble de contrôles pouvant être activés ou désactivés de manière groupée à l'aide d'une case à cocher unique. Les types de contrôle suivants peuvent être contenus dans un groupe de cases à cocher : sélecteur de champ, case à cocher, zone de liste

déroulante, zone de liste, contrôle de texte, contrôle de nombre, texte statique, groupe de boutons d'option et navigateur de fichiers. Le contrôle Groupe de cases à cocher possède les propriétés suivantes :

**Identificateur.** Identificateur unique du contrôle. Identificateur à utiliser pour référencer le contrôle dans le modèle de script.

**Titre.** Titre facultatif pour le groupe. Pour les titres comportant plusieurs lignes, utilisez \n pour indiquer les sauts de ligne.

**Titre de case à cocher.** Libellé facultatif qui s'affiche avec la case à cocher de contrôle. Prend en charge \n pour spécifier les sauts de ligne.

**Infobulle.** Texte d'infobulle facultatif qui apparaît lorsque l'utilisateur survole le contrôle.

**Touche mnémotechnique.** Caractère facultatif dans le titre à utiliser comme raccourci clavier pour activer le contrôle. Il est souligné dans le titre. Le raccourci est activé en appuyant sur Alt+[touche mnémotechnique].

**Valeur par défaut.** L'état par défaut de la case à cocher de contrôle, c'est-à-dire coché ou non coché.

**Script R vérifié/non vérifié** Indique le script R qui est généré lorsque le contrôle est sélectionné et non sélectionné. Pour inclure le script R dans le modèle de script, utilisez la valeur de la propriété Identificateur. Le script R généré, à partir de la propriété Script R vérifié ou Script R non vérifié, est inséré aux emplacements spécifiés de l'identificateur. Par exemple, si l'identificateur est *checkboxgroup1*, lors de l'exécution, les instances de *%checkboxgroup1%* dans le modèle de script sont remplacées par la valeur de la propriété Script R vérifié lorsque la case est cochée et par la valeur de la propriété Script R non vérifié lorsque la case est non cochée.

- Vous pouvez spécifier tout script R ou Python for Spark valide. Pour les scripts multilignes ou les scripts longs, cliquez sur le bouton des points de suspension (...) et entrez votre script dans la boîte de dialogue Propriétés de script.
- Vous pouvez inclure des identificateurs pour les contrôles contenus dans le groupe de cases à cocher. Lors de l'exécution, les identificateurs sont remplacés par le script R généré par les contrôles.
- La valeur *%ThisValue%* peut être utilisée dans la propriété Script R vérifié ou Script R non vérifié. Elle génère une liste séparée par un espace de chaque script R généré par chacun des contrôles dans le groupe de cases à cocher, dans l'ordre d'apparition dans le groupe (du haut vers le bas).
- Par défaut la propriété Script R vérifié possède la valeur *%ThisValue%* et la propriété Script R non vérifié est vide.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle. La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.

## Navigateur de fichiers

Le contrôle Navigateur de fichiers est composé d'une zone de texte devant contenir un chemin d'accès à un fichier et un bouton de navigation qui ouvre une boîte de dialogue IBM SPSS Modeler standard permettant d'ouvrir ou d'enregistrer un fichier. Le contrôle Navigateur de fichiers possède les propriétés suivantes :

**Identificateur.** Identificateur unique du contrôle. Identificateur à utiliser pour référencer le contrôle dans le modèle de script.

**Titre.** Titre facultatif qui apparaît au-dessus du contrôle. Pour les titres comportant plusieurs lignes, utilisez \n pour indiquer les sauts de ligne.

**Position du titre.** Indique la position du titre du contrôle. Les valeurs sont Top (Haut) et Left (Gauche) (Top est la valeur par défaut).

**Infobulle.** Texte d'infobulle facultatif qui apparaît lorsque l'utilisateur survole le contrôle.

**Touche mnémotique.** Caractère facultatif dans le titre à utiliser comme raccourci clavier pour activer le contrôle. Il est souligné dans le titre. Le raccourci est activé en appuyant sur Alt+[touche mnémotique].

**Opération d'un fichier système.** Indique si la boîte de dialogue lancée par le bouton de navigation est adaptée à l'ouverture ou à l'enregistrement de fichiers. La valeur Ouvrir indique que la boîte de dialogue Parcourir valide l'existence du fichier spécifié. La valeur Enregistrer indique que la boîte de dialogue Parcourir ne valide pas l'existence du fichier spécifié.

**Type de navigateur.** Indique si la boîte de dialogue Parcourir est utilisée pour sélectionner un fichier (Localiser le fichier) ou pour sélectionner un dossier (Localiser le dossier).

**Filtre de fichier.** Cliquez sur le bouton points de suspension (...) pour ouvrir la boîte de dialogue Filtre à fichiers qui vous permet de spécifier les types de fichiers disponibles pour la boîte de dialogue Ouvrir ou Enregistrer. Par défaut, tous les types de fichier sont autorisés. Vous pouvez également ouvrir la boîte de dialogue Filtre de fichier en cliquant deux fois sur le contrôle Navigateur de fichiers dans le canevas.

**Type de fichier système.** En mode d'analyse réparti, cela indique si la boîte de dialogue d'ouverture ou d'enregistrement parcourt le système de fichiers dans lequel s'exécute IBM SPSS Modeler Server ou le système de fichier de votre ordinateur local. Sélectionnez **Serveur** pour parcourir le système de fichiers du serveur ou **Client** pour parcourir le système de fichiers de votre ordinateur local. Cette propriété n'a aucun effet en mode d'analyse local.

**Requis pour l'exécution.** Indique si une valeur est requise dans ce contrôle pour que l'exécution puisse continuer. Si **True** est spécifié, l'utilisateur de la boîte de dialogue de noeud doit indiquer une valeur pour le contrôle, sinon une erreur est générée lorsqu'il clique sur le bouton **OK**. Si **False** est spécifié, l'absence de valeur dans ce contrôle n'a aucune incidence sur l'état du bouton **OK**. La valeur par défaut est **False**.

**Par défaut.** Valeur par défaut du contrôle.

**Script.** Spécifie le script R ou Python for Spark généré par ce contrôle lors de l'exécution et pouvant être inséré dans le modèle de script.

- Vous pouvez spécifier tout script R ou Python for Spark valide. Pour les scripts multilignes ou les scripts longs, cliquez sur le bouton des points de suspension (...) et entrez votre script dans la boîte de dialogue Propriétés de script.
- La valeur `%%ThisValue%%` indique la valeur d'exécution de la zone de texte, qui est le chemin de fichier entouré de doubles points d'interrogation, spécifié manuellement ou rempli automatiquement par la boîte de dialogue Parcourir. Il s'agit de la valeur par défaut.
- Si la propriété Script inclut `%%ThisValue%%` et que la valeur d'exécution de la zone de texte est vide, le contrôle de navigateur de fichiers ne génère pas de script.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle. La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.

**Filtre de type de fichier :** La boîte de dialogue Filtre de fichier permet de spécifier les types de fichier affichés dans les listes déroulantes Fichiers de type et Enregistrer sous le type pour ouvrir et enregistrer des boîtes de dialogue accessibles à partir d'un contrôle Navigateur de fichiers système. Par défaut, tous les types de fichier sont autorisés.

Pour spécifier des types de fichier non répertoriés explicitement dans la boîte de dialogue :

1. Sélectionnez Autre.
2. Entrez un nom pour le type de fichier.
3. Entrez un type de fichier au format `*.suffix`, par exemple `*.xls`. Vous pouvez spécifier plusieurs types de fichier, séparés par un point-virgule.

## Onglet

Le contrôle Onglet (Tab) ajoute un nouvel onglet à la boîte de dialogue de noeud. Tous les autres contrôles peuvent être ajoutés au nouvel onglet. Le contrôle Onglet possède les propriétés suivantes :

**Identificateur.** Identificateur unique du contrôle.

**Titre.** Titre de l'onglet.

**Position.** Indique la position de l'onglet dans la boîte de dialogue de noeud, par rapport aux autres onglets qui composent cette dernière.

**Script.** Indique le script R ou Python for Spark qui est généré et exécuté par ce contrôle lors de l'exécution et qui peut être inséré dans le modèle de script.

- Vous pouvez spécifier n'importe quel script R ou Python for Spark valide et utiliser `\n` pour les sauts de ligne.
- La valeur `%%ThisValue%%` génère une liste séparée par des espaces des scripts R ou Python for Spark générés par chaque contrôle dans l'onglet, dans l'ordre d'apparition dans l'onglet (du haut vers le bas et de gauche à droite). Il s'agit de la valeur par défaut.
- Si la propriété Script inclut `%%ThisValue%%` et qu'aucun script R ou Python for Spark n'est généré par les contrôles dans l'onglet, ce dernier ne génère pas de script.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle. La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.

## Bouton de la sous-boîte de dialogue

Le contrôle Bouton de la sous-boîte de dialogue spécifie un bouton de lancement d'une sous-boîte de dialogue et donne accès au générateur de boîtes de dialogue de la sous-boîte de dialogue. Le bouton Sous-boîte de dialogue possède les propriétés suivantes :

**Identificateur.** Identificateur unique du contrôle.

**Titre.** Texte qui apparaît dans le bouton.

**Infobulle.** Texte d'infobulle facultatif qui apparaît lorsque l'utilisateur survole le contrôle.

**Sous-boîte de dialogue.** Cliquez sur le bouton des points de suspension (...) afin d'ouvrir le générateur de boîtes de dialogue personnalisées pour la sous-boîte de dialogue. Vous pouvez également ouvrir le générateur en cliquant deux fois sur le bouton Sous-boîte de dialogue.

**Touche mnémonique.** Caractère facultatif dans le titre à utiliser comme raccourci clavier pour activer le contrôle. Il est souligné dans le titre. Le raccourci est activé en appuyant sur `Alt+[touche mnémonique]`.

**Règle d'activation.** Indique une règle qui détermine le moment auquel le contrôle en cours est activé. Cliquez sur le bouton des points de suspension (...) pour ouvrir la boîte de dialogue et spécifier la règle. La propriété Règle d'activation n'est visible que si d'autres contrôles qui peuvent être utilisés pour spécifier une règle d'activation existent sur le canevas.



**Remarque :** Le contrôle Bouton de boîte de dialogue de niveau inférieur ne peut pas être ajoutée à une boîte de dialogue de niveau inférieur.

**Propriétés d'une sous-boîte de dialogue :** Pour afficher et définir les propriétés d'une sous-boîte de dialogue :

1. Ouvrez la sous-boîte de dialogue en double-cliquant sur le bouton de la boîte de dialogue de niveau inférieur dans la boîte de dialogue principale ou cliquez une fois sur le bouton de la sous-boîte de dialogue et cliquez sur les points de suspension (...) de la propriété Sous-boîte de dialogue.
2. Dans la sous-boîte de dialogue, cliquez sur le canevas dans la zone située en dehors des contrôles. Sans contrôle sur le canevas, les propriétés d'une sous-boîte de dialogue sont toujours visibles.

**Nom de sous-boîte de dialogue.** Identificateur unique de la sous-boîte de dialogue. La propriété Sous-boîte de dialogue est obligatoire.

**Remarque :** Si vous spécifiez Nom de sous-boîte de dialogue comme identificateur dans le modèle de script, comme dans `%%My Sub-dialog Name%%`, il est remplacé lors de l'exécution par une liste séparée par des espaces du script généré par chaque contrôle dans la sous-boîte de dialogue, dans l'ordre d'apparition (du haut vers le bas et de gauche à droite).

**Titre.** Spécifie le texte à afficher dans la barre de titre de la sous-boîte de dialogue. La propriété Titre est facultative mais il est recommandé de la définir.

**Fichier d'aide.** Indique le chemin d'accès à un fichier d'aide facultatif pour la sous-boîte de dialogue. Il s'agit du fichier qui est lancé lorsque l'utilisateur clique sur le bouton **Aide** dans la sous-boîte de dialogue et il peut être identique au fichier d'aide indiqué pour la boîte de dialogue principale. Les fichiers d'aide doivent être au format HTML. Pour plus d'informations, voir la description de la propriété Fichier d'aide pour Propriétés des boîtes de dialogue.

### **Définition d'une règle d'activation pour un contrôle**

Vous pouvez définir une règle qui détermine les conditions d'activation d'un contrôle. Par exemple, la règle peut activer un groupe de boutons radio lorsqu'un sélecteur de champ est alimenté. Les options disponibles pour la définition de la règle d'activation dépendent du type de contrôle qui définit la règle.

#### **Sélecteur de champ**

Vous pouvez choisir d'activer un contrôle lorsqu'un sélecteur de champ contient au moins une zone (n'est pas vide). Vous pouvez aussi choisir de l'activer lorsque le sélecteur de champ n'est pas rempli(est vide).

#### **Case à cocher ou Groupe de cases à cocher**

Vous pouvez choisir d'activer un contrôle lorsqu'une case ou un groupe de cases est coché. Vous pouvez choisir de l'activer lorsque la case ou le groupe de cases n'est pas coché.

#### **Zone combinée ou Zones de liste à sélection unique**

Vous pouvez choisir d'activer un contrôle lorsqu'une valeur particulière est sélectionnée dans une zone combinée ou dans une zone de liste à sélection unique. Vous pouvez aussi choisir de l'activer lorsque la valeur n'est pas sélectionnée dans la zone combinée ou dans la zone de liste à sélection unique.

#### **Zone de liste à sélection multiple**

Vous pouvez choisir d'activer un contrôle lorsqu'une valeur particulière fait partie des valeurs sélectionnées d'une zone de liste à sélection multiple. Vous pouvez aussi choisir de l'activer lorsque la valeur ne fait pas partie des valeurs sélectionnées de la zone de liste à sélection multiple.

#### **Groupe de boutons radio**

Vous pouvez choisir d'activer un contrôle lorsqu'un bouton radio particulier est sélectionné. Vous pouvez aussi choisir de l'activer lorsque le bouton radio n'est pas sélectionné.

La propriété Règle d'activation est associée aux contrôles pour lesquels une règle d'activation peut être définie.

**Remarque :**

- Les règles d'activation s'appliquent indépendamment de l'activation du contrôle qui les définissent. Soit, par exemple, une règle spécifiant qu'un groupe de boutons radio est activé lorsqu'un sélecteur de champ est alimenté. Le groupe de boutons est activé dès que le sélecteur de champ est alimenté, que sélecteur de champ soit activé ou non.
- Lorsqu'un contrôle d'onglet est désactivé, tous les contrôles qu'il contient sont désactivés, même si les conditions de la règle d'activation de certains d'entre eux sont réunies.
- Lorsqu'un groupe de cases à cocher est désactivé, tous les contrôles du groupe sont désactivés, même si la case qui les contrôle est cochée.

## Propriétés d'extension

La boîte de dialogue Propriétés d'extension spécifie les informations sur l'extension en cours dans le générateur de boîtes de dialogue personnalisées pour les extensions, telles que le nom de l'extension et les fichiers dans l'extension.

- Toutes les boîtes de dialogue de noeud personnalisées créées dans le générateur de boîtes de dialogue personnalisées pour les extensions font partie d'une extension.
- Les champs de l'onglet Obligatoire de la boîte de dialogue Propriétés d'extension doivent être spécifiés pour que vous puissiez installer une extension et les boîtes de dialogue de noeud qui s'y trouvent.

Pour spécifier les propriétés d'une extension, dans les menus du générateur de boîtes de dialogue personnalisées pour les extensions, choisissez :

### Extension > Propriétés

#### Propriétés requises des extensions

**Nom** Nom unique à associer à l'extension. Il peut comporter jusqu'à trois mots et n'est pas sensible à la casse. Les caractères sont limités à l'ASCII à sept bits. Pour réduire la possibilité de conflits de noms, vous pouvez utiliser un nom comportant plusieurs mots, où le premier mot est un identificateur de votre société, tel qu'une URL.

#### Récapitulatif

Breve description de l'extension qui doit s'afficher sur une seule ligne.

#### Version

Identificateur de version au format x.x.x, dont composant doit être un entier (par exemple, 1.0.0). Les zéros sont implicites s'ils ne sont pas indiqués. Par exemple, l'identificateur de version 3.1 signifie 3.1.0. L'identificateur de version est indépendant de la version d'IBM SPSS Modeler.

#### Version minimale d'IBM SPSS Modeler

La version minimum d'IBM SPSS Modeler nécessaire à l'exécution de l'extension.

#### Fichiers

La liste Fichiers affiche les fichiers actuellement inclus dans l'extension. Cliquez sur **Ajouter** pour ajouter des fichiers à l'extension. Vous pouvez également supprimer des fichiers de l'extension et extraire des fichiers dans un dossier spécifié.

- Les boîtes de dialogue de noeud personnalisées possèdent le type de fichier .cfe.
- Les fichiers de traduction des composants de l'extension sont ajoutés à partir des paramètres Localisation de l'onglet Facultatif.
- Vous pouvez ajouter un fichier Readme à l'extension. Nommez le fichier ReadMe.txt. Les utilisateurs peuvent alors accéder au fichier Readme à partir de la boîte de dialogue qui affiche

les détails de l'extension. Vous pouvez inclure des versions traduites des fichiers Readme, appelés ReadMe\_<identificateur de langue>.txt, comme par exemple ReadMe\_fr.txt pour une version française.

## Propriétés facultatives des extensions

### Propriétés générales

#### Description

Une description plus détaillée de l'extension que celle fournie dans le champ **Récapitulatif**. Par exemple, vous pouvez répertorier les fonctions principales disponibles dans l'extension.

**Date** Date facultative pour la version actuelle de l'extension. Aucun formatage nécessaire.

#### Auteur

Auteur de l'extension. Vous pouvez inclure une adresse électronique.

**Liens** Ensemble d'URL à associer à l'extension (par exemple, la page d'accueil de l'auteur). Le format de ce champ est arbitraire, aussi veuillez délimiter plusieurs URL avec des espaces, des virgules ou d'autres délimiteurs pertinents.

#### Mots clés

Ensemble de mots clés auquel associer l'extension.

#### Plateforme

Informations sur les restrictions qui s'appliquent à l'utilisation de l'extension sur des plateformes de système d'exploitation particulières.

## Dépendances

### Maximum SPSS Modeler Version (Version maximale de SPSS Modeler)

Version maximale d'IBM SPSS Modeler sur laquelle l'extension peut être exécutée.

### Integration Plug-in for R requis

Indique si Integration Plug-in for R est requis.

Si l'extension requiert l'un des packs R du référentiel de packs CRAN, entrez les noms de ces packs dans le contrôle Packs R nécessaires. Les noms sont sensibles à la casse. Pour ajouter le premier pack, cliquez n'importe où sur le contrôle Packs R nécessaires afin de mettre en évidence le champ de saisie. Appuyez sur **Entrée**, le curseur se trouvant sur une ligne donnée, pour créer une nouvelle ligne. Vous pouvez supprimer une ligne en la sélectionnant et en appuyant sur **Suppr.**

## Localisation

### Noeuds personnalisés

Vous pouvez ajouter des versions traduites du fichier de propriétés (spécifie toutes les chaînes qui apparaissent dans la boîte de dialogue de noeud) pour une boîte de dialogue de noeud personnalisée dans l'extension. Pour ajouter des traductions pour une boîte de dialogue de noeud particulière, sélectionnez cette boîte de dialogue, cliquez sur **Add Translations** (Ajouter des traductions), puis sélectionnez le dossier qui contient les versions traduites. Tous les fichiers traduits d'une boîte de dialogue de noeud particulière doivent se trouver dans le même dossier. Pour des instructions sur la création des fichiers traduits, reportez-vous à la rubrique «Création de versions localisées des boîtes de dialogue de noeud personnalisées», à la page 57.

### Dossier des catalogues de traduction

Vous pouvez fournir des versions localisées des champs **Récapitulatif** et **Description** associés à l'extension qui s'affichent lorsque les utilisateurs visualisent les détails de l'extension à partir du Hub d'extension. L'ensemble des fichiers localisés pour une extension doit être placé dans un dossier intitulé lang. Accédez au dossier lang qui contient les fichiers localisés et sélectionnez ce dossier.

Pour fournir des versions localisées des champs **Récapitulatif** et **Description**, créez un fichier nommé `<nom_extension>_<identificateur_langue>.properties` pour chaque langue pour laquelle une traduction est proposée. Au moment de l'exécution, si le fichier `.properties` associé à la langue de l'interface utilisateur en cours est introuvable, les valeurs des champs **Récapitulatif** et **Description** spécifiés dans les onglets Nécessaire et Facultatif sont utilisées.

- `<nom_extension>` désigne la valeur du champ **Nom** associé à l'extension dans laquelle tous les espaces sont remplacés par un trait de soulignement.
- `<identificateur_langue>` représente l'identificateur de langue d'une langue spécifique. Les identificateurs de langues prises en charge par IBM SPSS Modeler sont affichés comme suit.

Par exemple, les traductions françaises d'une extension nommée MYORG MYSTAT sont stockées dans le fichier `MYORG_MYSTAT_fr.properties`.

Le fichier `.properties` doit contenir les deux lignes suivantes, qui indiquent le texte localisé pour les deux champs :

```
Summary=<texte localisé pour le champ Récapitulatif>
Description=<texte localisé pour le champ Description>
```

- Les mots clés `Summary` et `Description` doivent apparaître en anglais et le texte localisé doit se trouver sur la même ligne que le mot clé, sans retour à la ligne.
- Le fichier doit être au format de codage ISO 8859-1. Les caractères qui ne sont pas directement compatibles avec ce codage doivent être écrits avec des espaces Unicode ("`\u`").

Le dossier `lang` dans lequel sont enregistrés les fichiers localisés doit posséder un sous-dossier nommé `<identificateur-langue>` qui contient le fichier `.properties` localisé pour une langue spécifique. Par exemple, le fichier français `.properties` doit se trouver dans le dossier `lang/fr`.

Identificateurs de langue

**de.** Allemand

**en.** Anglais

**es.** Espagnol

**fr.** Français

**it.** Italien

**ja.** japonais

**ko.** Coréen

**pl.** Polonais

**pt\_BR.** Portugais (Brésil)

**ru.** Russe

**zh\_CN.** Chinois simplifié

**zh\_TW.** Chinois traditionnel

## Gestion des boîtes de dialogue de noeud personnalisées

Le générateur de boîtes de dialogue personnalisées pour les extensions permet de gérer les boîtes de dialogue de noeud personnalisées, dans des extensions créées par vous ou d'autres utilisateurs. Les boîtes

de dialogue de noeud personnalisées doivent être installées dans toutes les instances de SPSS Modeler Client ou SPSS Modeler Batch, où elles sont nécessaires pour pouvoir être utilisées. Notez qu'aucun élément ne doit être installé dans SPSS Modeler Server pour pouvoir utiliser un noeud de boîtes de dialogue personnalisées en mode serveur.

**Remarque :** Vous pouvez modifier les boîtes de dialogue de noeud personnalisées créées dans IBM SPSS Modeler.

## **Ouverture d'une extension qui contient des boîtes de dialogue de noeud personnalisées**

Vous pouvez ouvrir un fichier (.mpe) de bundle d'extension qui contient les spécifications d'une ou de plusieurs boîtes de dialogue de noeud personnalisées ou ouvrir une extension installée. Vous pouvez modifier l'une des boîtes de dialogue de noeud dans l'extension et enregistrer ou installer l'extension. L'installation de l'extension installe les boîtes de dialogue de noeud contenues dans l'extension. L'enregistrement de l'extension enregistre les modifications apportées aux boîtes de dialogue de noeud de l'extension.

Pour ouvrir un fichier de bundle d'extension, dans les menus du générateur de boîtes de dialogue personnalisées pour les extensions, choisissez :

### **Fichier > Ouvrir**

Pour ouvrir une extension installée, dans les menus du générateur de boîtes de dialogue personnalisées pour les extensions, choisissez :

### **Fichier > Ouvrir installé**

**Remarque :** Si vous ouvrez une extension installée pour la modifier, l'option **Fichier > Installer** la réinstalle en remplaçant la version existante. L'utilisation de l'option **Editer** dans le menu contextuel d'un noeud créé à l'aide du générateur de boîtes de dialogue personnalisées n'ouvre pas la boîte de dialogue de noeud dans le générateur de boîtes de dialogue personnalisées.

## **Enregistrement dans un fichier de bundle d'extension**

L'enregistrement de l'extension ouverte dans le générateur de boîtes de dialogue personnalisées pour les extensions enregistre également les boîtes de dialogue de noeud personnalisées de l'extension. Les extensions sont enregistrées dans un fichier de bundle d'extension (.mpe).

Dans les menus du Générateur de boîtes de dialogue personnalisées pour extensions, choisissez :

### **Fichier > Enregistrer**

## **Installation d'une extension**

L'installation de l'extension ouverte dans le générateur de boîtes de dialogue personnalisées pour les extensions installe également les boîtes de dialogue de noeud personnalisées de l'extension. L'installation d'une extension existante remplace la version existante, ce qui inclut le remplacement de toutes les boîtes de dialogue de noeud de l'extension qui étaient déjà installées.

Pour installer l'extension actuellement ouverte, dans les menus du générateur de boîtes de dialogue personnalisées pour les extensions, choisissez :

### **Fichier > Installer**

Par défaut, les extensions sont installées dans un emplacement général auquel l'utilisateur a accès en écriture pour votre système d'exploitation. Pour plus d'informations, voir la rubrique «Emplacements de l'installation des extensions», à la page 26.

**Remarque :** Dans un flux ouvert, les versions existantes des boîtes de dialogue de noeud contenues dans l'extension ne seront pas remplacées. Lorsque vous ouvrez un flux contenant un noeud Générateur de boîtes de dialogue personnalisées qui a été réinstallé, vous recevez un message d'avertissement.

## Désinstallation d'une extension

Dans les menus du Générateur de boîtes de dialogue personnalisées pour extensions, choisissez :

**Fichier > Désinstaller**

La désinstallation d'une extension désinstalle toutes les boîtes de dialogue de noeud contenues dans l'extension. Vous pouvez également désinstaller des extensions du Hub d'extension.

## Importation d'un fichier de pack de boîte de dialogue personnalisée

Vous pouvez importer un fichier (.cfd) de pack de boîte de dialogue personnalisée dans le générateur de boîtes de dialogue personnalisées pour les extensions. Le fichier .cfd est converti en fichier .cfe, qui est ajouté à la nouvelle extension.

Dans les menus du Générateur de boîtes de dialogue personnalisées pour extensions, choisissez :

**Fichier > Importer**

Vous pouvez également ajouter des fichiers .cfe à une extension à partir de la boîte de dialogue Propriétés d'extension, accessible depuis **Extension > Propriétés** dans le générateur de boîtes de dialogue personnalisées pour les extensions.

## Ajout d'une boîte de dialogue de noeud personnalisée à une extension

Vous pouvez ajouter une nouvelle boîte de dialogue de noeud personnalisée à une extension.

Dans les menus du Générateur de boîtes de dialogue personnalisées pour extensions, choisissez :

**Extension > Nouvelle boîte de dialogue**

## Permutation entre plusieurs boîtes de dialogue de noeud personnalisées dans une extension

Si l'extension actuelle contient plusieurs boîtes de dialogue de noeud personnalisées, vous pouvez passer de l'une à l'autre.

Dans les menus du Générateur de boîtes de dialogue personnalisées pour extensions, choisissez :

**Extension > Edit Dialog (Editer une boîte de dialogue)** et sélectionnez la boîte de dialogue de noeud personnalisée que vous souhaitez utiliser.

## Création d'une extension

Lorsque vous créez le générateur de boîtes de dialogue personnalisées pour les extensions, une nouvelle boîte de dialogue de noeud personnalisée vide est ajoutée à l'extension.

Pour créer une extension, dans les menus du générateur de boîtes de dialogue personnalisées pour les extensions, choisissez :

Fichier > Nouveau

## Extensions dans SPSS Modeler Batch ou IBM SPSS Collaboration and Deployment Services

Pour utiliser des extensions dans une installation SPSS Modeler Batch ou IBM SPSS Collaboration and Deployment Services, vérifiez que la variable d'environnement *IBM\_SPSS\_MODELER\_EXTENSION\_PATH* est définie dans l'environnement cible et qu'elle pointe vers l'emplacement qui contient les extensions. Si les flux qui contiennent un noeud R ont été stockés dans IBM SPSS Collaboration and Deployment Services Repository avant la définition de la variable d'environnement *IBM\_SPSS\_MODELER\_EXTENSION\_PATH*, vous devez restocker les flux vers le référentiel pour qu'ils puissent s'exécuter.

**Remarque :** Vérifiez que la version de l'adaptateur SPSS Modeler Batch ou IBM SPSS Collaboration and Deployment Services pour SPSS Modeler correspond à celle de SPSS Modeler Client où l'extension a été créée.

## Création de versions localisées des boîtes de dialogue de noeud personnalisées

Vous pouvez créer des versions localisées des boîtes de dialogue de noeud personnalisées pour n'importe quelle langue prise en charge par IBM SPSS Modeler. Vous pouvez localiser n'importe quelle chaîne apparaissant dans une boîte de dialogue de noeud personnalisée, ainsi que le fichier d'aide facultatif.

### Pour localiser des chaînes de boîte de dialogue

Vous devez créer une copie du fichier de propriétés associé à la boîte de dialogue de noeud personnalisée pour chaque langue que vous prévoyez de déployer. Ce fichier contient toutes les chaînes localisables associées à la boîte de dialogue de noeud.

Extrayez le fichier (.cfe) de la boîte de dialogue de noeud de l'extension en sélectionnant ce fichier dans la boîte de dialogue Propriétés d'extension (dans le générateur de boîtes de dialogue personnalisées pour les extensions) et en cliquant sur **Extraire**. Extrayez ensuite le contenu du fichier .cfe. Un fichier .cfe est simplement un fichier .zip. Le contenu extrait d'un fichier .cfe inclut un fichier de propriétés pour chaque langue prise en charge, le nom du fichier d'une langue particulière étant indiqué par <nom\_boîte\_dialogue>\_<identificateur\_langue>.properties (voir les identificateurs de langue du tableau qui suit).

1. Ouvrez chaque fichier de propriétés que vous prévoyez de traduire, à l'aide d'un éditeur de texte prenant en charge UTF-8, comme le Bloc-notes sous Windows. Modifiez les valeurs associées aux propriétés qui doivent être localisées, mais pas les noms des propriétés. Les propriétés associées à un contrôle spécifique sont préfixées avec l'identificateur du contrôle. Par exemple, la propriété ToolTip d'un contrôle avec l'identificateur *options\_button* est *options\_button\_tooltip\_LABEL*. Les propriétés de titre sont simplement nommées <identificateur>\_LABEL, comme dans *options\_button\_LABEL*.
2. Rajoutez les versions localisées des fichiers de propriétés dans le fichier (.cfe) de la boîte de dialogue de noeud personnalisée à partir des paramètres de localisation de l'onglet Facultatif de la boîte de dialogue Propriétés d'extension. Pour plus d'informations, reportez-vous à la rubrique «Propriétés facultatives des extensions», à la page 53.

Lorsque la boîte de dialogue de noeud est lancée, IBM SPSS Modeler recherche un fichier de propriétés dont l'identificateur de langue correspond à la langue en cours, comme indiqué par la liste déroulante Langue dans l'onglet Général de la boîte de dialogue Options. Si aucun fichier de propriétés n'est trouvé, le fichier par défaut <Nom de la boîte de dialogue>.properties est utilisé.

## Pour localiser le fichier d'aide

1. Générez une copie du fichier d'aide associé à la boîte de dialogue de noeud personnalisée et localisez le texte pour la langue voulue.
2. Renommez la copie en <Fichier d'aide>\_<identificateur de langue>.htm, en utilisant les identificateurs de langue présentés dans le tableau ci-dessous. Par exemple, si le fichier d'aide est myhelp.htm et que vous voulez créer une version allemande du fichier, le fichier d'aide localisé doit être nommé myhelp\_de.htm.

Stockez toutes les versions localisées du fichier d'aide dans le même répertoire que la version non localisée. Lorsque vous ajoutez le fichier d'aide non localisé à partir de la propriété Fichier d'aide des propriétés de la boîte de dialogue, les versions localisées sont automatiquement ajoutées à la boîte de dialogue de noeud.

S'il existe des fichiers supplémentaires, comme des fichiers images qui doivent eux aussi être localisés, vous devez modifier manuellement les chemins appropriés dans le fichier d'aide principal de sorte qu'ils pointent vers les versions localisées. Les fichiers supplémentaires, y compris les versions localisées, doivent être ajoutés manuellement au fichier (.cfe) de la boîte de dialogue de noeud personnalisée. pour plus d'informations sur l'accès aux fichiers de la boîte de dialogue de noeud personnalisée et à la modification manuelle de ces fichiers, reportez-vous à la section précédente intitulée "Pour localiser des chaînes de boîte de dialogue".

Lorsque la boîte de dialogue de noeud est lancée, IBM SPSS Modeler recherche un fichier d'aide dont l'identificateur de langue correspond à la langue en cours, comme indiqué par la liste déroulante Langue dans l'onglet Général de la boîte de dialogue Options. Si ce fichier d'aide n'est pas trouvé, le fichier d'aide spécifié pour la boîte de dialogue de noeud (fichier indiqué dans la propriété Fichier d'aide des Propriétés des boîtes de dialogue) est utilisé.

Identificateurs de langue

**de.** Allemand

**en.** Anglais

**es.** Espagnol

**fr.** Français

**it.** Italien

**ja.** japonais

**ko.** Coréen

**pl.** Polonais

**pt\_BR.** Portugais (Brésil)

**ru.** Russe

**zh\_CN.** Chinois simplifié

**zh\_TW.** Chinois traditionnel

*Remarque* : Le texte dans les boîtes de dialogue de noeud personnalisées et les fichiers d'aide associés n'est pas limité aux langues prises en charge par IBM SPSS Modeler. Vous êtes libre d'écrire la boîte de



dialogue de noeud et le texte d'aide dans n'importe quelle langue sans créer des fichiers de propriétés et des fichiers d'aide dans cette langue. Tous les utilisateurs de votre boîte de dialogue de noeud voient alors le texte dans cette langue.

## Importation et exportation de données à l'aide de Python for Spark

A l'aide du générateur de boîtes de dialogue personnalisées pour les extensions, vous pouvez créer des noeuds personnalisés et écrire des scripts Python for Spark pour lire les données de votre source de données, où qu'elle se trouve, et écrire des données dans tout format pris en charge par Apache Spark.

Par exemple, un utilisateur souhaite écrire ses données dans une base de données. Il utilise le générateur de boîtes de dialogue personnalisées pour les extensions et Python for Spark pour créer un noeud JDBC d'exportation personnalisé, puis il exécute le modèle pour écrire les données dans une base de données. Pour lire les données de la base de données, il peut également créer un noeud JDBC d'importation personnalisé. Il pourrait utiliser cette même méthode pour lire les données de SPSS Modeler à partir d'un fichier JSON, par exemple. Ensuite, après avoir lu ses données dans SPSS Modeler, il peut utiliser tous les noeuds SPSS Modeler disponibles pour travailler sur son problème métier.

**Remarque :** Si vous souhaitez utiliser JDBC avec la fonctionnalité d'importation et d'exportation de Python for Spark, vous devez copier votre fichier de pilote JDBC dans le répertoire `as/lib`, dans votre répertoire d'installation IBM SPSS Modeler.

## Importation et exportation de données à l'aide de Python for Spark

1. Accédez à **Extensions > Générateur de boîtes de dialogue de noeud personnalisées**.
2. Sous les propriétés des boîtes de dialogue, sélectionnez **Python for Spark** pour Type de script et **Importation** ou **Exportation** pour Type de noeud.
3. Entrez d'autres propriétés si nécessaire, telles que Nom de la boîte de dialogue.
4. Dans la section Script, entrez ou collez votre script Python for Spark pour l'importation ou l'exportation de données.
5. Cliquez sur **Installer** pour installer le script Python for Spark. Les nouveaux noeuds d'importation personnalisés seront ajoutés à la palette Sources et les nouveaux noeuds d'exportation personnalisés, à la palette Exportation.

## Importation et exportation de données à l'aide de R

A l'aide du générateur de boîtes de dialogue personnalisées pour les extensions, vous pouvez créer des noeuds personnalisés et écrire des scripts R pour lire les données de votre source de données, où qu'elle se trouve, et écrire des données dans tout format pris en charge par R.

Par exemple, un utilisateur souhaite écrire ses données dans une base de données. Il utilise le générateur de boîtes de dialogue personnalisées pour les extensions et le scriptage R pour créer un noeud JDBC d'exportation personnalisé, puis il exécute le modèle pour écrire les données dans une base de données. Pour lire les données de la base de données, il peut également créer un noeud JDBC d'importation personnalisé. Il pourrait utiliser cette même méthode pour lire les données de SPSS Modeler à partir d'un fichier JSON, par exemple. Ensuite, après avoir lu ses données dans SPSS Modeler, il peut utiliser tous les noeuds SPSS Modeler disponibles pour travailler sur son problème métier.

## Importation et exportation de données à l'aide de R

1. Accédez à **Extensions > Générateur de boîtes de dialogue de noeud personnalisées**.
2. Sous les propriétés des boîtes de dialogue, sélectionnez **R** pour Type de script et **Importation** ou **Exportation** pour Type de noeud.
3. Entrez d'autres propriétés si nécessaire, telles que Nom de la boîte de dialogue.
4. Dans la section Script, entrez ou collez votre script R pour l'importation ou l'exportation de données.

5. Cliquez sur **Installer** pour installer le script R. Les nouveaux noeuds d'importation personnalisés seront ajoutés à la palette Sources et les nouveaux noeuds d'exportation personnalisés, à la palette Exportation.

---

## Remarques

Ce document peut être disponible dans d'autres langues auprès d'IBM. Toutefois, il peut être nécessaire de posséder une copie du produit ou de la version du produit dans cette langue pour pouvoir y accéder.

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans ce pays. Pour plus de détails, référez-vous aux documents d'annonce disponibles dans votre pays ou adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service IBM puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets. Vous pouvez envoyer par écrit des questions concernant la licence à :

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

Pour toute demande au sujet des licences concernant les jeux de caractères codés sur deux octets (DBCS), contactez le service Propriété intellectuelle IBM de votre pays ou adressez vos questions par écrit à :

*Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan Ltd.  
19-21, Nihonbashi-Hakozakicho, Chuo-ku  
Tokyo 103-8510, Japan*

LE PRESENT DOCUMENT EST LIVRE "EN L'ETAT" SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFACON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites dans certaines transactions, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Le présent document peut contenir des inexactitudes ou des coquilles. Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut, à tout moment et sans préavis, modifier les produits et logiciels décrits dans ce document.

Les références à des sites Web non IBM sont fournies à titre d'information uniquement et n'impliquent en aucun cas une adhésion aux données qu'ils contiennent. Les éléments figurant sur ces sites Web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

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

Ces informations peuvent être disponibles, soumises à des conditions générales, et dans certains cas payantes.

Le logiciel sous licence décrit dans ce document et tous les éléments sous licence disponibles s'y rapportant sont fournis par IBM conformément aux dispositions de l'IBM Customer Agreement, des Conditions internationales d'utilisation des logiciels IBM ou de tout autre accord équivalent.

Les données de performances et les exemples de clients ne sont présentés qu'à des fins d'illustration. Les performances réelles peuvent varier en fonction des configurations et des conditions d'exploitation spécifiques.

Les informations concernant les produits autres qu'IBM ont été obtenues auprès des fabricants de ces produits, de leurs annonces publiques ou d'autres sources publiques disponibles. IBM n'a pas testé ces produits et ne peut confirmer l'exactitude de leurs performances ni leur compatibilité. Elle ne peut recevoir aucune réclamation concernant des produits non IBM. Toute question concernant les performances de produits non IBM doit être adressée aux fournisseurs de ces produits.

Toute instruction relative aux intentions d'IBM pour ses opérations à venir est susceptible d'être modifiée ou annulée sans préavis, et doit être considérée uniquement comme un objectif.

Le présent document peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs de personnes, de sociétés, de marques ou de produits à des fins illustratives ou explicatives uniquement. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite.

---

## **Marques**

IBM, le logo IBM et [ibm.com](http://ibm.com) sont des marques d'International Business Machines Corp., aux Etats-Unis et/ou dans certains autres pays. D'autres noms de produit et de service peuvent être des marques d'IBM ou d'autres sociétés. La liste actualisée de toutes les marques d'IBM est disponible sur la page Web "Copyright and trademark information" à l'adresse [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Adobe, le logo Adobe, PostScript et le logo PostScript sont des marques d'Adobe Systems Incorporated aux Etats-Unis et/ou dans certains autres pays.

Intel, le logo Intel, Intel Inside, le logo Intel Inside, Intel Centrino, le logo Intel Centrino, Celeron, Intel Xeon, Intel SpeedStep, Itanium, et Pentium sont des marques commerciales ou des marques déposées de Intel Corporation ou de ses filiales aux Etats-Unis et dans d'autres pays.

Linux est une marque déposée de Linus Torvalds aux Etats-Unis et/ou dans d'autres pays.

Microsoft, Windows, Windows NT et le logo Windows sont des marques commerciales de Microsoft Corporation aux Etats-Unis et/ou dans d'autres pays.

UNIX est une marque déposée de The Open Group aux Etats-Unis et dans d'autres pays.

Les marques commerciales Java et basées sur Java ainsi que les logos sont des marques commerciales ou déposées de Oracle et/ou de ses filiales.

---

## **Dispositions relatives à la documentation du produit**

Les droits d'utilisation relatifs à ces publications sont soumis aux dispositions suivantes.

### **Conditions d'utilisation**

Les présentes dispositions viennent s'ajouter à toute autre condition d'utilisation applicable au site Web IBM.

### **Usage personnel**

Vous pouvez reproduire ces publications pour un usage personnel et non commercial, sous réserve que toutes les mentions de propriété soient conservées. Vous ne pouvez distribuer ou publier tout ou partie de ces publications ou en faire des oeuvres dérivées sans le consentement exprès d'IBM.

### **Usage commercial**

Vous pouvez reproduire, distribuer et afficher ces publications uniquement au sein de votre entreprise, sous réserve que toutes les mentions de propriété soient conservées. Vous ne pouvez reproduire, distribuer, afficher ou publier tout ou partie de ces publications en dehors de votre entreprise, ou en faire des oeuvres dérivées, sans le consentement exprès d'IBM.

### **Droits**

Excepté les droits d'utilisation expressément accordés dans ce document, aucun autre droit, licence ou autorisation, implicite ou explicite, n'est accordé pour ces publications ou autres informations, données, logiciels ou droits de propriété intellectuelle contenus dans ces publications.

IBM se réserve le droit de retirer les autorisations accordées dans la présente, à sa discrétion, si l'utilisation des publications s'avère préjudiciable à ses intérêts ou que, selon son appréciation, les instructions susmentionnées n'ont pas été respectées.

Vous ne pouvez télécharger, exporter ou réexporter ces informations qu'en total accord avec toutes les lois et règlements applicables dans votre pays, y compris les lois et règlements américains relatifs à l'exportation.

IBM N'OCTROIE AUCUNE GARANTIE SUR LE CONTENU DE CES PUBLICATIONS. LES PUBLICATIONS SONT LIVREES EN L'ETAT SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES PUBLICATIONS EN CAS DE CONTREFAÇON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE.



---

# Index

## A

Apache Spark 1

## B

bundles d'extension  
installation de bundles  
d'extension 26

## E

extensions 23  
affichage des extensions installées 25  
détails de l'extension 26  
installation des mises à jour des  
extensions 24  
recherche et installation de nouvelles  
extensions 24  
suppression d'extensions 25  
extensions IBM SPSS Modeler 1

## F

fichier .par 13  
fichier .pim 13

## G

générateur de boîtes de dialogue  
personnalisées  
activation des règles 51  
aperçu 32  
bouton de sous-boîte de dialogue 50  
case à cocher 34  
colonnes des contrôles de tableau 44  
contrôle de date 40  
contrôle de texte 38  
contrôle des nombres 39  
contrôle Groupe d'éléments 46  
contrôle table 43  
contrôle Texte statique 42  
éléments de liste 36  
éléments de liste déroulante 36  
exportation de données Python for  
Spark 59  
exportation de données R 59  
fichier d'aide 29  
filtrage des listes de champs 34  
filtre de type de fichier 49  
groupe de boutons radio 46, 47  
groupe de cases à cocher 47  
importation de données Python for  
Spark 59  
importation de données R 59  
localisation des boîtes de dialogue et  
des fichiers d'aide 57  
modèle de script 31  
navigateur de fichiers 48  
onglet 50

générateur de boîtes de dialogue  
personnalisées (*suite*)  
propriétés de la boîte de dialogue 29  
propriétés de sous-boîte de  
dialogue 51  
règles de présentation 30  
sélecteur de champ 33  
sélecteur de couleur 43  
source de la zone 34  
texte sécurisé 41  
zone de liste 37  
zone de liste déroulante 35  
générateur de boîtes de dialogue  
personnalisées pour les extensions 27  
enregistrement d'extensions contenant  
des boîtes de dialogue de noeud 54  
fichiers de bundle d'extension 54  
installation d'extensions contenant des  
boîtes de dialogue de noeud 54  
modification de boîtes de dialogue de  
noeud dans les extensions  
installées 54  
ouverture d'extensions contenant des  
boîtes de dialogue de noeud 54

## I

IBM SPSS Modeler Solution  
Publisher 13

## N

navigateur Sortie d'extension 16  
Navigateur Sortie d'extension 16  
noeud de transformation d'extension 20  
onglet Sortie de la console 20  
noeud Exportation d'extension 12  
onglet Sortie de la console 13  
noeud Importation d'extension 21  
onglet Sortie de la console 21  
noeud Modèle d'extension 16  
onglet Options de modèle 17  
onglet Sortie de la console 17  
onglet Sortie texte 17  
noeud Sortie d'extension 14  
onglet Sortie 15  
onglet Sortie de la console 15  
onglet Syntaxe 14  
Noeuds d'extension 12  
noeuds d'extension IBM SPSS  
Modeler 12  
nugget de modèle Extension 17  
à propos de 17  
Onglet Options de modèle 18  
onglet Sortie de graphique 19  
onglet Sortie de la console 19  
onglet Sortie de texte 19  
onglet Syntaxe 18

## O

onglet Sortie de graphique  
nugget de modèle Extension 19  
onglet Sortie de la console  
nugget de modèle Extension 19  
onglet Sortie de texte  
nugget de modèle Extension 19  
onglet Sortie texte 16  
onglet Syntaxe  
nugget de modèle Extension 18

## P

publication de flux  
IBM SPSS Modeler Solution  
Publisher 13  
Python for Spark 1  
API 2, 4, 7, 8, 9

## R

R  
syntaxe admise 1

## S

sortie graphique (onglet) 16  
syntaxe R admise 1









Imprimé en France