

IBM SPSS Modeler Extensions



Nota

Antes de utilizar essas informações e o produto que elas suportam, leia as informações em “[Avisos](#)” na [página 63](#).

Informações do produto

Esta edição se aplica à versão 18, release 4, modificação 0 de IBM® SPSS Modeler e a todos os lançamentos e modificações subsequentes até indicado de outra forma em novas edições.

© Copyright International Business Machines Corporation .

Índice

Capítulo 1. Idiomas suportados.....	1
R.....	1
Python for Spark.....	1
Script com Python for Spark.....	2
Alternando para um ambiente Python diferente.....	12
APIs nativas do Python.....	12
Nós de Extensão.....	14
Nó de Exportação de extensão.....	14
Nó de saída de extensão.....	16
Nó de modelo de extensão.....	19
Nugget do modelo de extensão.....	20
Nó de transformação de extensão.....	22
Nó de importação de extensão.....	24
Capítulo 2. Extensões.....	27
Hub de extensão.....	27
Guia Explorar.....	28
Guia Instaladas.....	29
Configurações.....	29
Detalhes de Extensão.....	30
Instalando pacotes configuráveis de extensão.....	30
Locais de instalação para extensões.....	30
Pacotes R necessários.....	31
Criando e gerenciando nós customizados.....	31
Layout do construtor de diálogo personalizado.....	32
Construindo um diálogo de nó customizado.....	33
Propriedades do diálogo.....	33
Efetuando layout em controle na tela de diálogo	34
Construindo o modelo de script.....	34
Visualizando um diálogo de nó customizado.....	36
Tipos de controle.....	36
Propriedades de Extensão.....	54
Gerenciando diálogos nó customizado.....	57
Criando versões localizadas dos diálogos de nó customizados.....	59
Importando e exportando dados usando Python para Spark.....	60
Importando e exportando dados usando R.....	61
Avisos.....	63
Marcas comerciais.....	64
Termos e condições para documentação do produto.....	64
Índice remissivo.....	67

Capítulo 1. Idiomas suportados

IBM SPSS Modeler suporta R e Apache Spark (via Python). Consulte as informações a seguir para obter mais informações.

R

O IBM SPSS Modeler suporta R.

Sintaxe R permitida

- No campo de sintaxe na guia **Sintaxe** dos vários nós de Extensão, são permitidas apenas instruções e funções que são reconhecidas por R.
- Para o nó de transformação extensão e o nugget do modelo Extensão, os dados passam pelo script R (em lote). Por esta razão, scripts R para pontuação de modelo e nós de processo não devem incluir operações que abrangem ou combinam linhas nos dados, como classificação ou agregação. Essa limitação é imposta para garantir que os dados possam ser divididos em um ambiente Hadoop e durante a mineração dentro da base de dados. Os nós de saída de extensão e de construção de modelo de extensão não têm essa limitação.
- A adição de um modo de transferência de dados sem lote, em ambos o nó de Extensão Transformar e o nugget do modelo de Extensão, significa que você pode se estender ou combinar linhas nos dados em Servidor do SPSS Modeler.
- Todos os nós R podem ser vistos como ambientes R globais independentes. Portanto, o uso de funções `library` dentro dos dois nós R separados requer o carregamento da biblioteca R em ambos os scripts R.
- Para exibir o valor de um objeto R definido em seu script R, deve-se incluir uma chamada em uma função de impressão. Por exemplo, para exibir o valor de um objeto R chamado de `data`, inclua as seguintes linhas em seu script R:

```
print(data)
```

- Não é possível incluir uma chamada na função `setwd` de R em seu script R, pois essa função é usada pelo IBM SPSS Modeler para controlar o caminho do arquivo do arquivo de saída dos scripts R.
- Os parâmetros do fluxo definidos para uso nas expressões do CLEM e em script não são reconhecidos se usados em scripts R.
- O IBM SPSS Modeler não suporta o plot interativo em R
- Alguns objetos R são preenchidos automaticamente quando nós de Extensão são usados em um fluxo (por exemplo, `modelerData` e `modelerDataModel` quadros de dados).

Python for Spark

O IBM SPSS Modeler suporta scripts Python for Apache Spark.

Nota:

- Os nós Python dependem do ambiente Spark.
- Os scripts Python devem usar a API do Spark porque os dados serão apresentados na forma de um Spark DataFrame.
- Os nós antigos criados na versão 17.1 ainda serão executados com relação a Servidor analítico IBM SPSS (os dados são originados de um nó de origem Servidor analítico IBM SPSS e não foram extraídos para o IBM SPSS Modeler Server). Os novos nós do Python e do Construtor de Diálogo Customizado criados na versão 18.0 ou posterior podem ser executados no IBM SPSS Modeler Server.

- Ao instalar o Python, certifique-se de que todos os usuários tenham permissão para acessar a instalação do Python.
- Se você desejar usar a Biblioteca do Machine Learning (MLlib), deverá instalar uma versão do Python que inclua NumPy. Em seguida, deve-se configurar o IBM SPSS Modeler Server (ou o servidor local no IBM SPSS Modeler Client) para usar sua instalação do Python . Para obter detalhes, consulte “[Script com Python for Spark](#)” na página 2.

Script com Python for Spark

O IBM SPSS Modeler pode executar scripts Python usando a estrutura Apache Spark para processar dados. Esta documentação fornece a descrição da API do Python para as interfaces fornecidas.

A instalação ' IBM SPSS Modeler inclui uma distribuição do Spark (por exemplo, ' IBM SPSS Modeler ' 18.6 inclui o Spark 3.4.0).

Pré-Requisitos

- Se você planeja executar scripts Python em " Servidor analíticoIBM SPSS, deve ter uma conexão com o Analytic Server, e o Analytic Server deve ter acesso a uma instalação compatível do Apache Spark. Consulte a documentação do " Servidor analíticoIBM SPSS para obter detalhes sobre o uso do Apache Spark como mecanismo de execução.
- Se você planeja executar scripts Python no servidor ' IBM SPSS Modeler (ou no servidor local incluído no cliente ' IBM SPSS Modeler, que requer Windows 64 ou Mac64), não será mais necessário instalar Python e editar o ' options .cfg para usar sua instalação Python. A partir da versão 18.1, o ' IBM SPSS Modeler agora inclui uma distribuição Python. No entanto, se você precisar de um determinado módulo que não esteja incluído na distribuição padrão Python " IBM SPSS Modeler, vá para " <Modeler_installation_directory>/python e instale pacotes extras.

Embora uma distribuição Python esteja agora incluída com ' IBM SPSS Modeler, você ainda pode apontar para sua própria instalação Python, como nas versões anteriores, se desejar, adicionando a seguinte opção a ' options .cfg:

```
# Set to the full path to the python executable
(including the executable name) to enable use of PySpark.
eas_pyspark_python_path, ""
```

Exemplo do Windows:

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

Exemplo de Linux:

```
eas_pyspark_python_path, "/Your_Python_Install/bin/python"
```

Nota: Se você apontar para sua própria instalação Python, ela deverá ser da versão 3.8.x. IBM SPSS Modeler foi testado com o Anaconda para Python 3.8 e Python 3.8.6.

- Se você planeja determinar a versão PySpark que é usada pelo ' IBM SPSS Modeler, pode executar o seguinte script.

```
import pkg_resources
print("pandas version")
print(pkg_resources.get_distribution("pandas").version)
print("pyspark version")
print(pkg_resources.get_distribution("pyspark").version)
```

Com base em seu sistema operacional, execute os seguintes comandos nos caminhos correspondentes antes de executar o script para determinar a versão PySpark.

Para Windows

Caminho: '<Modeler-InstallationDirectory>\18.6\spark\python

Comando:

```
"<Modeler-InstallationDirectory>\18.6\python\python.exe" setup.py sdist
```

Para Mac

Caminho: '*<Modeler-InstallationDirectory>/18.6/IBM SPSS Modeler.app/Contents/spark/python*

Comando:

```
"<Modeler-InstallationDirectory>/18.6/IBM SPSS Modeler.app/Contents/python/bin/python3" setup.py sdist
```

Para Linux

Caminho: '*<Modeler-InstallationDirectory>/18.6/spark/python*

Comando:

```
<Modeler-InstallationDirectory>/18.6/python/bin/python3 setup.py sdist
```

Nota: *<Modeler-InstallationDirectory>* corresponde à instalação do Thick Client ' IBM SPSS Modeler para o servidor local, enquanto corresponde à instalação do servidor ' IBM SPSS Modeler para o servidor remoto.

O objeto de contexto ' Servidor analíticoIBM SPSS

O contexto de execução de um script Python é definido por um objeto de contexto do Analytic Server. Ao executar no servidor ' IBM SPSS Modeler, o objeto de contexto é para a versão incorporada do Analytic Server que está incluída na instalação do servidor ' IBM SPSS Modeler. Para obter o objeto de contexto, o script deve incluir o seguinte comando:

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

No contexto do Analytic Server, você pode obter o contexto do Spark e o contexto do SQL:

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

Consulte a documentação Apache Spark para obter informações sobre o contexto do Spark e o contexto do SQL.

Acessar dados

Os dados são transferidos entre um script Python/Spark e o contexto de execução na forma de um Spark SQL DataFrame. Um script que consome dados (ou seja, qualquer nó, exceto um nó de origem) deve recuperar o quadro de dados do contexto:

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

Um script que produz dados (ou seja, qualquer nó, exceto um nó terminal) deve retornar um quadro de dados para o contexto:

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

É possível usar o contexto SQL para criar um quadro de dados de saída de um RDD quando necessário:

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

Definindo o modelo de dados

Um nó que produz dados também deve definir um modelo de dados que descreve os campos visíveis de recebimento de dados do nó. Na terminologia do Spark SQL, o modelo de dados é o esquema.

Um script Python/Spark define seu modelo de dados de saída na forma de um objeto `pyspark.sql.types.StructType`. Um `StructType` descreve uma linha no quadro de dados de saída e é construído a partir de uma lista de objetos `StructField`. Cada `StructField` descreve um único campo no modelo de dados de saída.

É possível obter o modelo de dados para os dados de entrada usando o atributo `:schema` do quadro de dados de entrada:

```
inputSchema = inputData.schema
```

Os campos que são passados inalterados podem ser copiados do modelo de dados de entrada para o modelo de dados de saída. Os campos que são novos ou modificados no modelo de dados de saída podem ser criados usando o construtor `StructField`:

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

Consulte a documentação do Spark para obter informações sobre o construtor.

Forneça pelo menos o nome do campo e seu tipo de dados. Opcionalmente, é possível especificar metadados para fornecer uma medida, função e descrição para o campo (consulte “[Metadados de dados](#)” na página 7).

Modo DataModelOnly

IBM SPSS Modeler precisa conhecer o modelo de dados de saída de um nó, antes que o nó seja executado, para permitir a edição downstream. Para obter o modelo de dados de saída de um nó Python, IBM SPSS Modeler executa o script em um modo especial “somente modelo de dados”, no qual nenhum dado está disponível. O script pode identificar esse modo usando o método `isComputeDataModelOnly` no objeto de contexto do Analytic Server.

O script para um nó de transformação pode seguir este padrão geral:

```
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)
```

Construindo um modelo

Um nó que constrói um modelo deve retornar ao contexto de execução algum conteúdo que descreve o modelo suficientemente para que o nó que aplica o modelo possa recriá-lo exatamente em um momento posterior.

O conteúdo do modelo é definido em termos de pares de chave/valor em que o significado das chaves e dos valores é conhecido apenas pelos nós de construção e pontuação e não é interpretado pelo Modelador de forma alguma. Opcionalmente, o nó pode atribuir um tipo MIME a um valor com a intenção de que o Modeler exiba os valores que têm tipos conhecidos para o usuário no nugget do modelo.

Um valor nesse contexto pode ser PMML, HTML, uma imagem e assim por diante. Para incluir um valor no conteúdo do modelo (no script de construção):

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

Para recuperar um valor do conteúdo do modelo (no script de pontuação):

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

Como atalho, quando um modelo ou parte de um modelo é armazenado em um arquivo ou pasta no sistema de arquivos, é possível agrupar todo o conteúdo armazenado nesse local em uma única chamada (no script de compilação):

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

Observe que, neste caso, não há opção para especificar um tipo MIME porque o pacote configurável pode conter vários tipos de conteúdo.

Se você precisar de um local temporário para armazenar o conteúdo durante a criação do modelo, poderá obter um local apropriado no contexto:

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

Para recuperar o conteúdo existente em um local temporário no sistema de arquivos (no script de pontuação):

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

Manipulação de erros

Para gerar erros, lance uma exceção do script e exiba-a para o usuário do IBM SPSS Modeler. Algumas exceções são predefinidas no módulo `spss.pyspark.exceptions`. Por exemplo:

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

Contexto do servidor analítico

O Contexto fornece suporte para a interface de contexto do Analytic Server para interação com Servidor analíticoIBM SPSS.

Objetos `AnalyticServerContext`

`AnalyticServerContext` Objetos configuram o ambiente de contexto que fornece várias interfaces para interagir com Servidor analíticoIBM SPSS. Um aplicativo que deseja construir essa instância de contexto deve fazer isso usando a interface `spss.pyspark.runtime.getContext()` em vez de implementar a interface diretamente.

Retorna a instância Pyspark python `SparkContext`:

```
ctx.getSparkContext() : SparkContext
```

Retorna a instância Pyspark python `SQLContext`:

```
ctx.getSparkSQLContext() : SQLContext
```

Retorna `True` para descrever se a execução é feita apenas para calcular o modelo de dados de saída. Caso contrário, retorna `False`:

```
ctx.isComputeModelOnly() : Boolean
```

Retorna `True` se o script estiver em execução no ambiente Spark. Atualmente, ele sempre retorna `True`:

```
ctx.isSparkExecution() : Boolean
```

Carrega dados de entrada do arquivo temporário upstream e gera a instância `pyspark.sql.DataFrame`:

```
ctx.getSparkInputData() : DataFrame
```

Retorna uma instância `pyspark.sql.StructType` gerada a partir do modelo de dados de entrada.
Retorna `None` se o modelo de dados de entrada não existir:

```
ctx.getSparkInputSchema() : StructType
```

Serializa o quadro de dados de saída em contexto do Analytic Server e retorna o contexto:

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

Parâmetro:

- `outDF (DataFrame)` : O valor do quadro de dados de saída

Exceções:

- `DataOutputNotSupported` : Se esta interface for invocada na função `pyspark:buildmodel`
- `ASContextException` : Se o quadro de dados de saída for `None`
- `InconsistentOutputDataModel` : Os nomes de campo e informações do tipo de armazenamento comuns a ambos os objetos é inconsistente

Converte a instância `outSchema StructType` em um modelo de dados, serializa-o no contexto do Analytic Server e retorna o contexto:

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

Parâmetro:

- `outSchema(StructType)` : A saída `StructType` objeto

Exceções:

- `ASContextException` : Se a instância do esquema de saída for `None`
- `InconsistentOutputDataModel` : Os nomes de campo e informações do tipo de armazenamento comuns a ambos os objetos é inconsistente

Armazena a localização da saída de construção de modelo para o contexto do Analytic Server e retorna o contexto:

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

O caminho pode ser um caminho de diretório que deve usar a API `ctx.createTemporaryFolder()` para gerar, quando tudo sob o diretório é empacotado como conteúdo de modelo.

Parâmetros:

- `key (string)` : valor da sequência de chaves
- `path (string)` : localização do caminho de cadeia de saída do prédio do modelo
- `mimetype (string, optional)` : o tipo MIME do conteúdo

Exceções:

- `ModelErrorNotSupported` : Quando não invoca essa API da função `pyspark:buildmodel`
- `KeyError` : Se o atributo chave for `None` ou a string estiver vazia

Armazena o conteúdo de construção de modelo, metadados ou outros atributos para o contexto do Analytic Server e retorna o contexto:

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

Parâmetros:

- `key (string)` : valor da sequência de chaves
- `value (string)` : o valor da cadeia de metadados do modelo
- `mimetype (string, optional)` : o tipo MIME do conteúdo

Exceções:

- ModelOutputNotSupported : Quando não invoca essa API da função pyspark:buildmodel
- KeyError : Se o atributo chave for None ou a string estiver vazia

Retorna o local da pasta temporária que é gerenciado pelo Analytic Server; isso pode ser usado para armazenar o conteúdo do modelo:

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

Exceção:

- ModelOutputNotSupported : Quando não invoca essa API da função pyspark:buildmodel

Retorna a localização do modelo que corresponde à chave de entrada:

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

Parâmetro:

- key (string) : valor da sequência de chaves

Exceções:

- ModelInputNotSupported : Quando não invoca essa API da função pyspark:applymodel
- KeyError : Se o atributo chave for None ou a string estiver vazia
- IncompatibleModelContentType : Se o tipo de conteúdo do modelo não for um container

Retorna o conteúdo do modelo, metadados do modelo ou outros atributos do modelo que correspondem à chave de entrada:

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

Parâmetro:

- key (string) : valor da sequência de chaves

Exceções:

- ModelInputNotSupported : Quando não invoca essa API da função pyspark:applymodel
- KeyError : Se o atributo chave for None, ou a string estiver vazia, ou a chave não existir
- IncompatibleModelContentType : Se o tipo de conteúdo do modelo não for consistente

Retorna o tipo MIME designado à chave de entrada. Ele retorna None se o conteúdo especificado não tiver tipo MIME:

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

Parâmetro:

- key (string) : valor da sequência de chaves

Exceções:

- ModelInputNotSupported : Quando não invoca essa API da função pyspark:applymodel
- KeyError : Se o atributo chave for None, ou a string estiver vazia, ou a chave não existir

Metadados de dados

Esta seção descreve como configurar os atributos do modelo de dados com base no pyspark.sql.StructField.

spss.datamodel.Role Objetos

Esta classe enumera funções válidas para cada campo em um modelo de dados.

BOTH: indica que este campo pode ser um antecedente ou um consequente.

FREQWEIGHT: Indica que este campo é usado para ser como peso de frequência; isto não é exibido para o usuário.

INPUT: indica que este campo é um preditor ou um antecedente.

NONE:indica que este campo não é usado diretamente durante a modelagem.

TARGET: indica que este campo é previsto ou subsequente.

PARTITION: Indica que este campo é usado para identificar a partição de dados.

RECORDID: Indica que este campo é usado para identificar o id do registro.

SPLIT: Indica que este campo é usado para dividir os dados.

spss.datamodel.Measure Objetos

Esta classe enumera níveis de medição para campos em um modelo de dados.

UNKNOWN: indica que o tipo de medida é desconhecido.

CONTINUOUS: indica que o tipo de medida é contínuo.

NOMINAL: indica que o tipo de medida é nominal.

FLAG: indica que o valor do campo é um de dois valores.

DISCRETE: indica que o valor do campo deve ser interpretado como uma coleção de valores.

ORDINAL: indica que o tipo de medida é ordinal.

TYPELESS: indica que o campo pode ter qualquer valor compatível com o seu armazenamento.

pyspark.sql.StructField Objetos

Representa um campo em um StructType. Um objeto StructField compreende quatro campos:

- name (string): nome de um StructField
- dataType (pyspark.sql.DataType): tipo de dado específico
- nullable (bool): se os valores de um StructField puderem conter valores None
- metadata (dictionary): um dicionário python usado para armazenar os atributos de opção

É possível usar a instância do dicionário de metadados para armazenar a medida, função ou atributo de rótulo para o campo específico. As palavras-chave para esses atributos são:

- measure: a palavra chave para o atributo measure
- role: a palavra chave para o atributo role
- displayLabel: a palavra chave para o atributo label

Exemplo:

```
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)
```

Data, horário, registro de data e hora

Para operações que utilizam data, hora ou dados do tipo timestamp, o valor é convertido para o valor real com base no valor 1970-01-01:00:00:00 (usando o Horário Universal Coordenado).

Para a data, o valor representa o número de dias, com base no valor 1970-01-01 (usando o Tempo Universal Coordenado).

Para o tempo, o valor representa o número de segundos às 24 horas.

Para o timestamp, o valor representa o número de segundos com base no valor 1970-01-01:00:00:00 (usando o Horário Universal Coordenado).

Exceções

Esta seção descreve possíveis instâncias de exceção.

Objetos MetadataException

Uma subclasse de Exceção python.

Esta exceção é lançada se ocorrer um erro durante operar o objeto de metadados.

UnsupportedOperationException Objetos

Uma subclasse de Exceção python.

Esta exceção é lançada se a operação específica não permitir a execução.

Objetos InconsistentOutputDataModel

Uma subclasse de Exceção python.

Essa exceção é lançada se `setSparkOutputSchema` e `setSparkOutputData` são chamados, mas os nomes de campo e as informações de tipo de armazenamento comuns a ambos os objetos são inconsistentes.

Objetos IncompatibleModelContentType

Uma subclasse de Exceção python.

Esta exceção é lançada durante os cenários a seguir:

- Usando `setModelContentFromString` para configurar o modelo mas usando `getModelContentToPath` para obter valor
- Usando `setModelContentFromPath` para configurar o modelo mas usando `getModelContentToString` para obter valor

Objetos DataOutputNotSupportedException

Uma subclasse de Exceção python.

Essa exceção é gerada em `setSparkOutputData` em uma execução tratada pela função `pyspark:buildmodel`.

Objetos ModelInputNotSupportedException

Uma subclasse de Exceção python.

Essa exceção só é levantada se o script não chamar a API `getModelContentPathByKey` e `getModelContentToString` na função `pyspark:applymodel`.

Objetos ModelOutputNotSupportedException

Uma subclasse de Exceção python.

Essa exceção só é levantada se o script não chamar as APIs `setModelContentFromPath` e `setModelContentFromString` na função `pyspark:buildmodel`.

Objetos ASContextException

Uma subclasse de Exceção python.

Essa exceção é lançada se ocorrer uma exceção de tempo de execução inesperada.

Exemplos

Esta seção contém Python para exemplos de scripting Spark.

Exemplo de script básico para processamento de dados

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

ctx = spss.pyspark.runtime.getContext()

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

Exemplo de script de construção de modelo, usando o algoritmo 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()

# field settings and algorithm parameters

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

# save linear regression model to a filesystem path

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

# print model details to 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]))

# check that required fields exist in the input data

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")

# define map function to convert from dataframe Row objects to 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))

# convert dataframe to an RDD containing LabeledPoint
training_points = df.rdd.map(lambda row:
row2LabeledPoint(target,predictors,row))

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

# write a text description of the model to stdout
dump(model,predictors)

# save the model to the filesystem and store into the output model content
modelpath = ascontext.createTemporaryFolder()
save(model,sc,modelpath)
ascontext.setModelContentFromPath("model",modelpath)

```

Exemplo de script de pontuação de modelo, usando o algoritmo 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%%]

# compute the output schema by adding the prediction field
outputSchema = ascontext.getSparkInputSchema()
outputSchema.fields.append(StructField(prediction_field,
DoubleType(), nullable=True))

# make a prediction based on a regression model and Dataframe Row object
# return a list containing the input row values and the predicted value
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

# load a serialized model from the filesystem

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():
    # score the data in the input data frame
    indf = ascontext.getSparkInputData()

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

    # compute the 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))

    # create an output DataFrame containing the scores
    sqlCtx = SQLContext(sc)
    outdf = sqlCtx.createDataFrame(scores_rdd, schema=outputSchema)

    # return the output DataFrame as the result
    ascontext.setSparkOutputData(outdf)

```

Alternando para um ambiente Python diferente

Você pode alterar o ambiente Python usado pelos nós de Extensão e Custom Dialog Builder em IBM SPSS Modeler:

1. Conceder permissões de edição para options.cfg:
 - a. A partir do diretório de instalação do IBM SPSS Modeler Server, abra config/options.cfg em um editor de texto.
 - b. Modifique a linha a seguir para conceder permissões de edição de usuários:

```
administrators, "*"
```

 - c. Salve o arquivo e, em seguida, reinicie o Servidor IBM SPSS Modeler .
2. Abra IBM SPSS Modeler e conecte-se ao IBM SPSS Modeler Server.
 3. Vá em **Ferramentas > Gerenciar Configuração do Servidor Modelador**.
 4. Em **Acesso ao Arquivo de Dados**, modifique o caminho executável Python e, em seguida, clique em **OK**.
 5. Reinicie o IBM SPSS Modeler Server e, em seguida, reconecte-se a ele em IBM SPSS Modeler.
 6. Vá em **Ferramentas > Gerenciar Configuração do Servidor Modelador** e verifique se o seu novo caminho Python foi salvo.

APIs nativas do Python

É possível chamar APIs nativas do Python de seus scripts para interagir com o IBM SPSS Modeler.

As APIs a seguir são suportadas

Para ver um exemplo, você pode baixar um [fluxo de amostra](#) e importá-lo para o IBM SPSS Modeler.

APIs para modelos de dados

- **modelerpy.isComputeDataModelOnly()**

É possível usar essa API para verificar se uma execução atual é calcular os dados de saída ou apenas calcular o modelo de dados de saída. Quando ele retorna true, seu script **não** deve executar nenhuma tarefa que dependa de dados de entrada ou de saída, caso contrário, a execução falhará

- **modelerpy.getDataModel()**

Essa API contata o IBM SPSS Modeler para obter o modelo de dados para um conjunto de dados de entrada. O valor de retorno é uma instância do class `DataModel` que descreve metadados do conjunto de dados de entrada, incluindo contagem de campo, nome do campo, tipo de armazenamento de campo, etc.

- **modelerpy.setOutputDataModel(dataModel)**

Essa API envia uma instância da classe DataModel de volta para IBM SPSS Modeler deve ser chamado antes que seu script transmita um conjunto de dados para IBM SPSS Modeler O IBM SPSS Modeler usará os metadados descritos nessa instância do DataModel para manipular seus dados no lado do IBM SPSS Modeler

APIs para modelagem

- **modelerpy.saveModel(model, name='model')**

Essa API transforma um modelo Python em um modelo IBM SPSS Modeler , que então é salvo pelo IBM SPSS Modeler. Você deve chamar essa API de um nó de modelagem quando um modelo Python for construído. Depois de chamar essa API, o modelo salvo é copiado para um nugget do modelo gerado

- **modelerpy.loadModel(name='model')**

Essa API carrega um modelo salvo do IBM SPSS Modeler e cria um objeto Python para o modelo salvo Chame esta API do nugget do modelo para carregar o modelo salvo para processamento adicional, como pontuação.

APIs para conjuntos de dados de entrada/saída

- **modelerpy.readPandasDataframe()**

Essa API lê um conjunto de dados do IBM SPSS Modeler para o Python O valor de retorno é um Python Pandas DataFrame (uma estrutura de dados bidimensional, como uma matriz bidimensional ou uma tabela com linhas e colunas).

- **modelerpy.writePandasDataframe(df)**

Essa API grava um Python Pandas DataFrame de Python para IBM SPSS Modeler

APIs para pacotes

- **modelerpy.installPackage(package)**

Essa API obtém um pacote do pypi.org e o instala.

- **modelerpy.uninstallPackage(package)**

Esta API desinstala um pacote instalado..

APIs para metadados

As classes relacionadas a metadados a seguir devem ser usadas com `modelerpy.getDataModel` e `modelerpy.setOutputDataModel`

- **modelerpy.DataModel**

Esta API é a classe de entrada principal para os metadados.. Ele contém uma matriz de instâncias de class Field e inclui os métodos a seguir

- **modelerpy.DataModel.getFields**

Esse método retorna a matriz de instâncias do class Field

- **modelerpy.DataModel.addField**

Este método inclui uma instância de Field para a matriz de metadados

- **modelerpy.Field**

A classe Field é onde as informações reais de metadados são armazenadas, incluindo o nome do campo, o armazenamento e a medição,

- **modelerpy.Field.getName**

Este método retorna o nome do campo..

- **modelerpy.Field.getStorage**

Este método retorna o armazenamento do campo.. Armazenamento válido inclui: integer, real, string, date, timee timestamp.

- **modelerpy.Field.getMeasure**

Este método retorna a medição do campo Medidas válidas incluem: discrete, flag, nominal, ordinal e continuous.

O código de exemplo a seguir constrói um objeto DataModel chamando o construtor modelerpy.DataModel com uma matriz modelerpy.Field. O construtor modelerpy.Field aceita o nome do campo, armazenamento de campo e medição de campo como seus parâmetros de entrada (armazenamento de campo e medição de campo são necessários; a medição de campo é opcional).

```
dataModel = modelerpy.DataModel([
    # %FieldName%, %StorageType%, %MeasurementType%
    modelerpy.Field('StringField', 'string', 'nominal'),
    modelerpy.Field('FloatField', 'real', 'continuous'),
    modelerpy.Field('IntegerField', 'integer', 'ordinal'),
    modelerpy.Field('BooleanField', 'integer', 'flag'),
    modelerpy.Field('DatetimeField', 'timestamp', 'continuous'),
    modelerpy.Field('TimeField', 'time', 'continuous'),
    modelerpy.Field('DateField', 'date', 'continuous'),
])
# StorageType could be: integer, real, string, date, time, timestamp
# MeasurementType could be: discrete, flag, nominal, ordinal, continuous

outputDataModel = modelerDataModel
outputDataModel.addField(modelerpy.Field(field_outlier, "real", measure="flag"))
outputDataModel.addField(modelerpy.Field(field_dist_hp, "real", measure="continuous"))
```

Nós de Extensão

Para complementar IBM SPSS Modeler e suas habilidades de mineração de dados, os nós de Extensão permitem que os usuários especialistas insiram seus próprios scripts R ou Python para scripts de Spark para realizar o processamento de dados, construção de modelo e pontuação de modelo.

Nó de Exportação de extensão

Com o nó Extensão Exportar, você pode executar R ou Python para scripts Spark para exportar dados.

Nó de Exportação de Extensão-Guia Sintaxe

Selecione seu tipo de sintaxe - **R** ou **Python for Spark**. Consulte as informações a seguir para obter mais informações. Quando a sua sintaxe estiver pronta, você pode clicar em **Executar** para executar o nó de Extensão Exportação.

Sintaxe de R

R Sintaxe. É possível inserir, ou colar, a sintaxe customizada do script R para a análise de dados nesse campo.

Converter campos de sinalização. Especifica como os campos de sinalização são tratados. Há duas opções: **Sequências para fator**, **Números inteiros e reais para dobro** e **Valores lógicos (True, False)**. Se você selecionar **Valores lógicos (True, False)** os valores originais dos campos de sinalização serão perdidos. Por exemplo, se um campo tiver os valores Masculino e Feminino, eles serão alterados para True e False.

Converter os valores ausentes para o valor R 'não disponível' (NA). Quando selecionado, quaisquer valores ausentes são convertidos para o valor NA R. O valor NA é usado por R para identificar valores ausentes. Algumas funções R que você usa podem ter um argumento que pode ser usado para controlar como a função se comporta quando os dados contêm NA. Por exemplo, a função pode permitir que você opte por excluir automaticamente os registros que contêm NA. Se essa opção não for selecionada,

qualquer valor omissos será passado para R inalterado e poderá causar erros quando o script R for executado.

Converter os campos data/hora em classes de R com controle especial para fusos horários. Quando selecionado, as variáveis com formatos de data ou data/hora são convertidas em objetos de data/hora de R. Você deve selecionar uma das seguintes opções:

- **Variáveis POSIXct.** R com formatos de data ou data e hora são convertidas em objetos POSIXct R.
- **R POSIXlt (lista).** Variáveis com formatos de data ou data e hora são convertidas em objetos POSIXlt R.

Nota: Os formatos POSIX são opções avançadas. Use essas opções somente se o script R especificar que os campos data/hora sejam tratados de uma forma que requeira esses formatos. Os formatos POSIX não se aplicam a variáveis com formatos de tempo.

Sintaxe do Python

Python Sintaxe. Você pode inserir, ou colar, personalizada sintaxe de script Python para análise de dados neste campo. Para obter mais informações sobre o Python para Spark, consulte [Python para Spark](#) e [Scripting com Python para Spark](#).

Nó de Extensão Exportação-Guia de Saída do

A guia **Saída do Console** contém qualquer saída que é recebida quando o script R ou Python para script Spark na guia Syntax executa (por exemplo, se usando um script R, ele mostra saída recebida do console R quando o script R no campo **R Syntax** na guia **Syntax** é executado). Essa saída pode incluir mensagens de erro R ou Python ou avisos que são produzidos quando o script R ou Python é executado. A saída pode ser usada, principalmente, para depurar o script. A guia **Saída de console** também contém o script a partir do campo **R Sintaxe** ou **Sintaxe Python**.

Toda vez que o script de Exportação de Extensão é executado, o conteúdo da guia **Saída do Console** é sobrescrito com a saída recebida do console R ou Python para Spark. A saída não pode ser editada.

fluxos de publicação

Os fluxos de publicação são feitos diretamente a partir de IBM SPSS Modeler usando qualquer um dos nós de exportação padrão: Banco de Dados, Arquivo Flat, Estatísticas Exportação, Exportação de Extensão, Exportação de Dados de Coleta de Dados, Exportação de SAS, Excel e XML Export. O tipo de nó de exportação determina o formato dos resultados a serem gravados todas as vezes em que o fluxo publicado for executado usando o Tempo de execução IBM SPSS Modeler Solution Publisher ou aplicativo externo. Por exemplo, se desejar gravar seus resultados em um banco de dados todas as vezes em que o fluxo publicado for executado, use um nó da Base de Dados.

Para publicar um fluxo

1. Abra ou construa um fluxo no modo normal e anexe um nó de exportação no término.
2. Na guia Publish no nó de exportação, especifique um nome de raiz para os arquivos publicados (ou seja, o nome do arquivo para o qual serão anexadas as extensões .pim, .parx .xml).
3. Clique em **Publicar** para publicar o fluxo, ou selecione **Publicar o fluxo** para publicar automaticamente o fluxo cada vez que o nó for executado.

Nome Publicado. Especificar o nome da raiz para os arquivos de imagem e parâmetro publicados.

- O **arquivo de imagem** (*.pim) fornece todas as informações necessárias para que o Tempo de Execução execute o fluxo publicado exatamente como foi no momento da exportação. Se você estiver confiante de que não precisará alterar nenhuma configuração para o fluxo (tal como a origem de dados de entrada ou o arquivo de dados de saída), é possível implementar apenas o arquivo de imagem.
- O **arquivo de parâmetro** (*.par) contém informações configuráveis sobre as origens de dados, arquivos de saída e opções de execução. Se você desejar poder controlar a entrada ou a saída do

fluxo sem publicar novamente o fluxo, você precisará tanto do arquivo de parâmetro quanto do arquivo de imagem.

- O **arquivo de metadados** (*.xml) descreve as entradas e saídas da imagem e seus modelos de dados. É projetado para ser usado por aplicativos que integram a biblioteca de tempo de execução e que precisam conhecer a estrutura dos dados de entrada e de saída.

Nota: Este arquivo só é produzido se você selecionar a opção **Metadados Publish**.

Publicar parâmetros. Se necessário, é possível incluir parâmetros de fluxo no arquivo *.par. É possível alterar esses valores de parâmetro de fluxo ao executar a imagem editando o arquivo *.par ou por meio da API de tempo de execução.

Esta opção possibilita o botão **Parâmetros**. A caixa de diálogo Parâmetros de Publicação é exibida ao clicar no botão.

Escolha os parâmetros que deseja incluir na imagem publicada, selecionando a opção relevante na coluna **Publish**.

Na execução de fluxo. Especifica se o fluxo é automaticamente publicado quando o nó for executado.

- **Dados de exportação.** Executa o nó de exportação no modo padrão, sem publicar o fluxo. (Basicamente, o nó executa em IBM SPSS Modeler da mesma forma que seria se Editor de soluçõesIBM SPSS Modeler não estivesse disponível.) Se você selecionar esta opção, o fluxo não será publicado, a menos que você faça isso explicitamente clicando em **Publicar** na caixa de diálogo do nó de exportação. Como alternativa, é possível publicar o fluxo atual usando a ferramenta Publicar na barra de ferramentas ou usando um script.
- **Publicar o fluxo.** Publica o fluxo para implementação usando o Editor de soluçõesIBM SPSS Modeler. Selecione essa opção se desejar publicar automaticamente o fluxo todas as vezes em que ele for executado.

Nota:

- Se você planeja executar o fluxo publicado com dados novos ou atualizados, é importante observar que a ordem dos campos no arquivo de entrada deve ser a mesma da ordem dos campos no arquivo de entrada do nó de origem especificada no fluxo publicado.
- Ao publicar em aplicativos externos, considere filtrar campos extras ou renomeados em conformidade com os requisitos de entrada. Ambos podem ser realizados usando um nó de Filtro antes do nó de exportação.

Nó de saída de extensão

Se **Saída para tela** for selecionada na guia **Output** do diálogo de nó da Extensão Output, saída em tela será exibida em uma janela do navegador de saída. A saída também será incluída no gerenciador de Saída. A janela do navegador de saída possui seu próprio conjunto de menus que permitem imprimir ou salvar a saída ou exportá-la em outro formato. O menu **Editar** contém somente a opção **Copiar**. O navegador de saída do nó de Extensão de Extensão possui duas guias; a guia **Output Output** que exibe saída de texto e a guia **Graph Output** que exibe gráficos e gráficos.

Se **Saída para arquivo** for selecionada na guia **Output** do diálogo de nó de Extensão de Extensão, a janela do navegador de saída não será exibida após a execução bem-sucedida do nó de Extensão de Extensão.

Nó de Saída de Extensão-Guia Sintaxe

Selecione seu tipo de sintaxe - **R** ou **Python for Spark**. Consulte as informações a seguir para obter mais informações. Quando a sua sintaxe estiver pronta, você pode clicar em **Executar** para executar o nó de Extensão Output. Os objetos de saída são incluídos no gerenciador de Saída, ou como opção, no arquivo especificado no campo **Nome do Arquivo** na guia **Saída**.

Sintaxe de R

R Sintaxe. É possível inserir, ou colar, a sintaxe customizada do script R para a análise de dados nesse campo.

Converter campos de sinalização. Especifica como os campos de sinalização são tratados. Há duas opções: **Sequências para fator, Números inteiros e reais para dobro** e **Valores lógicos (True, False)**. Se você selecionar **Valores lógicos (True, False)** os valores originais dos campos de sinalização serão perdidos. Por exemplo, se um campo tiver os valores Masculino e Feminino, eles serão alterados para True e False.

Converter os valores ausentes para o valor R 'não disponível' (NA). Quando selecionado, quaisquer valores ausentes são convertidos para o valor NA R. O valor NA é usado por R para identificar valores ausentes. Algumas funções R que você usa podem ter um argumento que pode ser usado para controlar como a função se comporta quando os dados contêm NA. Por exemplo, a função pode permitir que você opte por excluir automaticamente os registros que contêm NA. Se essa opção não for selecionada, qualquer valor omissso será passado para R inalterado e poderá causar erros quando o script R for executado.

Converter os campos data/hora em classes de R com controle especial para fusos horários. Quando selecionado, as variáveis com formatos de data ou data/hora são convertidas em objetos de data/hora de R. Você deve selecionar uma das seguintes opções:

- **Variáveis POSIXct.** R com formatos de data ou data e hora são convertidas em objetos POSIXct R.
- **R POSIXlt (lista).** Variáveis com formatos de data ou data e hora são convertidas em objetos POSIXlt R.

Nota: Os formatos POSIX são opções avançadas. Use essas opções somente se o script R especificar que os campos data/hora sejam tratados de uma forma que requeira esses formatos. Os formatos POSIX não se aplicam a variáveis com formatos de tempo.

Sintaxe do Python

Python Sintaxe. Você pode inserir, ou colar, personalizada sintaxe de script Python para análise de dados neste campo. Para obter mais informações sobre o Python para Spark, consulte [Python para Spark](#) e [Scripting com Python para Spark](#).

Nó de Saída de Extensão-Guia Saída do Console

A guia **Saída do Console** contém qualquer saída que é recebida quando o script R ou Python para script Spark na guia Syntax executa (por exemplo, se usando um script R, ele mostra saída recebida do console R quando o script R no campo **R Syntax** na guia **Syntax** é executado). Essa saída pode incluir mensagens de erro R ou Python ou avisos que são produzidos quando o script R ou Python é executado. A saída pode ser usada, principalmente, para depurar o script. A guia **Saída de console** também contém o script a partir do campo **R Sintaxe** ou **Sintaxe Python**.

Toda vez que o script de Saída de Extensão é executado, o conteúdo da guia **Saída do Console** é sobrescrito com a saída recebida do console R ou Python para Spark. A saída não pode ser editada.

Nó de Saída de Extensão-Guia Saída

Nome de saída. Especifica o nome da saída que é produzida quando o nó é executado. Quando **Auto** é selecionado, o nome da saída é configurado automaticamente para "R Output" ou "Python Output" dependendo do tipo de script. Opcionalmente, é possível selecionar **Customizado** para especificar um nome diferente.

Saída para a tela. Selecione essa opção para gerar e exibir a saída em uma nova janela. A saída também será incluída no gerenciador de Saída.

Saída para arquivo. Selecione essa opção para salvar a saída em um arquivo. Fazer isso ativa os botões de opção **Gráfico de Saída** e **Arquivo de Saída**.

Gráfico de Saída. Ativado somente se **Saída para arquivo** estiver selecionado. Assinale esta opção para salvar quaisquer gráficos que resultem da execução do nó de Extensão de Extensão em um arquivo.

Especifique um nome de arquivo a ser usado para a saída gerada no campo **Nome do arquivo**. Clique no botão de elipses (...) para escolher um arquivo e local específicos. Especifique o tipo de arquivo na lista suspensa **Tipo de arquivo**. Os seguintes tipos de arquivos estão disponíveis:

- Objeto de saída (.cou)
- HTML (.html)

Texto De Saída. Ativado somente se **Saída para arquivo** estiver selecionado. Selecione esta opção para salvar qualquer saída de texto que resulte em executar o nó de Extensão de Extensão em um arquivo.

Especifique um nome de arquivo a ser usado para a saída gerada no campo **Nome do arquivo**. Clique no botão de elipses (...) para especificar um arquivo e local específicos. Especifique o tipo de arquivo na lista suspensa **Tipo de arquivo**. Os seguintes tipos de arquivos estão disponíveis:

- HTML (.html)
- Objeto de saída (.cou)
- Documento de texto (.txt)

Navegador De Saída De Extensão

Se **Saída para tela** for selecionada na guia **Output** da caixa de diálogo de Extensão de Extensão, a saída em tela será exibida em uma janela do navegador de saída. A saída também será incluída no gerenciador de Saída. A janela do navegador de saída possui seu próprio conjunto de menus que permitem imprimir ou salvar a saída ou exportá-la em outro formato. O menu **Editar** contém somente a opção **Copiar**. O navegador de saída do nó de Extensão de Extensão possui duas guias:

- A guia **Saída de Texto** exibe saída de texto
- A guia **Graph Output** exibe gráficos e gráficos

Se **Saída para o arquivo** for selecionada na guia **Output** da caixa de diálogo de Extensão de Extensão em vez de **Output para tela**, a janela do navegador de saída não será exibida após a execução bem-sucedida do nó de Extensão de Extensão.

Guia De Saída De Extensão-Guia de Saída de Texto

A guia **Saída de Texto** exibe qualquer saída de texto gerada quando o script R ou Python para script Spark na guia **Sintaxe** do nó de Extensão Output é executado.

Nota: R ou Python para mensagens de erro de Spark ou avisos que resultam da execução do seu script de Output Output são sempre exibidos na guia **Saída do Console** do nó da Extensão Output.

Guia De Saída Browser-Guia Saída do gráfico

O nó Saída de Extensão para Python Spark agora inclui uma guia **Saída de Gráfico** como R. Essa guia exibe quaisquer gráficos ou diagramas que são gerados quando o script R ou o script Python para Spark na guia **Sintaxe** do nó Saída de Extensão é executado. Por exemplo, se o script R contiver uma chamada para a função plot de R, o gráfico resultante será exibido nessa guia. Agora é possível usar um script como o seguinte para gerar gráficos:

```
import spss.pyspark.runtime
import numpy
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

ascontext = spss.pyspark.runtime.getContext()
indf = ascontext.getSparkInputData()
sns.pairplot(indf.toPandas(), hue='K')
sns.pairplot(indf.toPandas(), hue='Age')
plt.show()
```

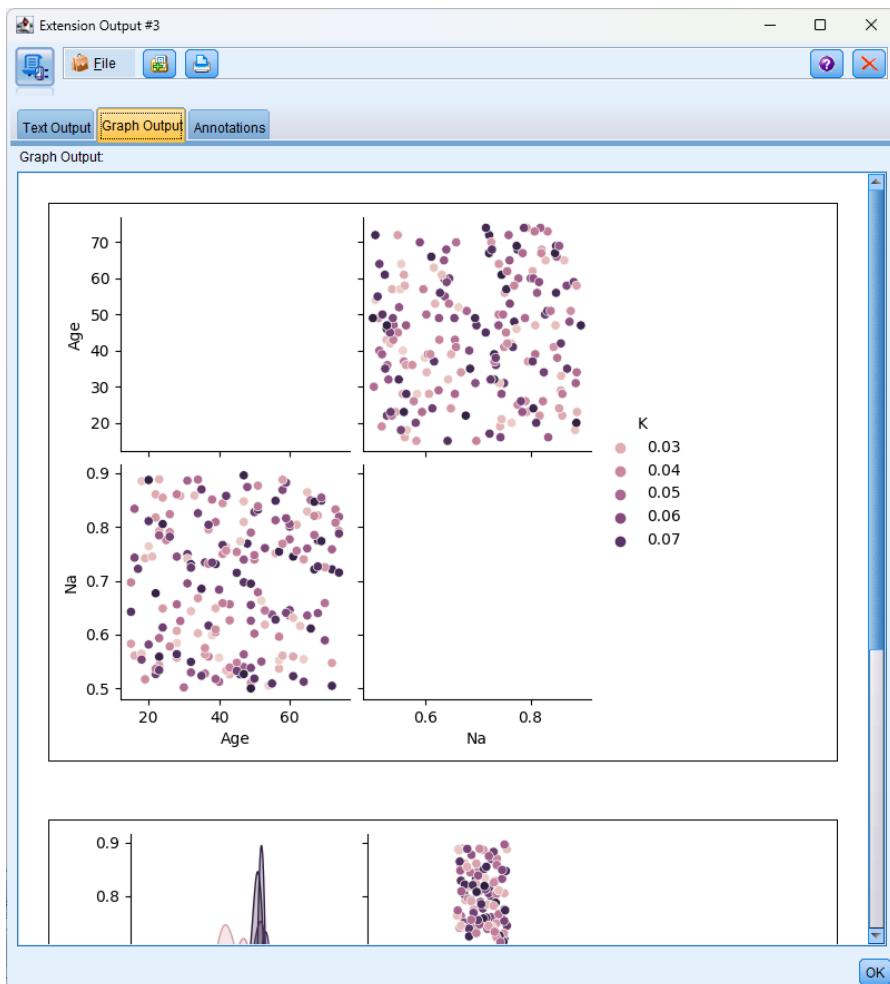


Figura 1. guia saída de gráfico

Nó de modelo de extensão

Com o nó do Modelo de Extensão, você pode executar o R ou Python para scripts Spark para construir e marcar modelos.

Nó do Modelo de Extensão-Guia Sintaxe

Selecione seu tipo de sintaxe - **R** ou **Python for Spark**. Em seguida, digite ou cole sua sintaxe de script personalizado em um dos campos a seguir. Quando a sua sintaxe estiver pronta, você pode clicar em **Executar** para executar o nó do Modelo de Extensão.

Sintaxe R

Sintaxe de construção do modelo R. É possível inserir, ou colar, a sintaxe customizada do script R para a construção do modelo nesse campo.

Sintaxe de pontuação do modelo R. É possível inserir, ou colar, a sintaxe customizada do script R para a escoragem do modelo nesse campo.

Python para sintaxe Spark

Python sintaxe de construção de modelo. Você pode inserir, ou colar, personalizada sintaxe de script Python para construção de modelo neste campo.

Sintaxe de pontuação do modelo Python. Você pode inserir, ou colar, personalizada sintaxe de script Python para modelagem de modelo neste campo.

Para obter mais informações sobre o Python para Spark, consulte “[Python for Spark](#)” na página 1 e “[Script com Python for Spark](#)” na página 2.

Nó do Modelo de Extensão-Guia Opções de Modelo

Nome do Modelo. Você pode gerar o nome do modelo automaticamente com base no campo de destino ou ID (ou tipo de modelo em casos em que nenhum campo é especificado), ou especificar um nome personalizado.

Nó do Modelo de Extensão-Guia Saída do Console

A guia **Saída do Console** contém qualquer saída que é recebida quando o script R ou Python para script Spark na guia Syntax executa (por exemplo, se usando um script R, ele mostra saída recebida do console R quando o script R no campo **R Syntax** na guia **Syntax** é executado). Essa saída pode incluir mensagens de erro R ou Python ou avisos que são produzidos quando o script R ou Python é executado. A saída pode ser usada, principalmente, para depurar o script. A guia **Saída de console** também contém o script a partir do campo **R Sintaxe** ou **Sintaxe Python**.

Toda vez que o script do Modelo de Extensão é executado, o conteúdo da guia **Saída do Console** é sobrescrito com a saída recebida do console R ou Python para Spark. A saída não pode ser editada.

Nó do Modelo de Extensão-Guia Saída de Texto

A guia **Saída de Texto** está presente no nó do Modelo de Extensão se solicitado selecionando a caixa de seleção **Exibir saída de texto do R** na guia **Opções do Modelo** da caixa de diálogo do nó do Modelo de Extensão. Essa guia pode exibir somente saída de texto. Uma saída de texto produzida ao executar o script de construção do modelo R é exibida nessa guia. Se você executar o script de construção de modelo novamente, sem primeiro ter especificado um nome diferente para o modelo, o conteúdo da guia **Saída de Texto** da execução anterior será sobrescrita. A saída de texto não pode ser editada.

Se você incluir uma chamada para a função `sink` de R em seu script, qualquer saída produzida após essa função será salva no arquivo especificado e não será exibida na guia **Saída de Texto**.

Nota: R ou Python para mensagens de erro de Spark ou avisos que resultam da execução do seu script de construção de modelo são sempre exibidos na guia **Output Console** do nó do Modelo de Extensão.

Nugget do modelo de extensão

O nugget de modelo de Extensão é gerado e colocado na paleta Models após executar o nó do Modelo de Extensão, que contém o script R ou Python para Spark que define o edifício do modelo e scoring do modelo. Por padrão, o nugget de modelo de Extensão contém o script usado para pontuação de modelo, opções para leitura dos dados e qualquer saída do console R ou Python para Spark. Opcionalmente, o nugget do modelo de extensão também pode conter várias outras formas de saída do modelo, como gráficos e saída de texto. Depois que o nugget de modelo de Extensão é gerado e adicionado à tela do fluxo, um nó de saída pode ser conectado a ele. O nó de saída é então usado na forma habitual nos fluxos do IBM SPSS Modeler para obter informações sobre os dados e modelos e para exportar dados em diversos formatos.

Para usar este nó com R, você deve instalar IBM SPSS Modeler - Essentials for R. Consulte o *IBM SPSS Modeler -Essentials for R: instruções de instalação* para obter instruções de instalação e informações de compatibilidade. Deve-se também ter uma versão compatível de R instalada no computador.

Nugget de modelo de extensão-Guia Sintaxe

A guia **Sintaxe** está sempre presente no nugget modelo de Extensão.

Sintaxe de pontuação do modelo R. Se usando R, o script R que é usado para pontuação de modelo é exibido neste campo. Por padrão, este campo é ativado mas não editável. Para editar o script de escoragem de modelo R, clique em **Editar**.

Sintaxe de pontuação do modelo Python. Se usando o Python para Spark, o script Python que é usado para pontuação de modelo é exibido neste campo. Por padrão, este campo é ativado mas não editável. Para editar o script de pontuação do modelo Python, clique em **Editar**.

Editar. Clique em **Editar** para fazer o campo de sintaxe scoring editável. Você pode então editar seu script de pontuação do modelo digitando no campo de sintaxe scoring. Por exemplo, você pode querer editar seu script de pontuação do modelo se você identificar um erro em seu script de pontuação de modelo após ter executado o nugget de modelo de Extensão. Quaisquer alterações que você fizer no script de pontuação do modelo no nugget modelo de Extensão serão perdidas se você regenerar o modelo executando o nó do Modelo de Extensão.

Nugget de modelo de extensão-Guia Opções de Modelo

A guia **Opções do Modelo** está sempre presente no nugget modelo de Extensão.

Leia as opções de dados. Essas opções se aplicam apenas ao R, não ao Python for Spark. Com essas opções, é possível especificar como os valores omissos, os campos de sinalização e as variáveis com formatos de data ou hora são manipulados.

- **Ler dados em lotes.** Se você estiver processando uma grande quantidade de dados (isso é muito grande para se encaixar na memória do motor R, por exemplo), use esta opção para quebrar os dados em lotes que podem ser enviados e processados individualmente. Especifique o número mínimo de registros de dados a serem incluídos em cada lote.

Tanto para o nó de Extensão quanto para o nugget de Extensão Scoring, os dados passam pelo script R (em lote). Por esta razão, scripts para pontuação de modelo e nós de processo que são executados em um ambiente de Hadoop ou Banco de Dados não devem incluir operações que abrangem ou combinam linhas nos dados, como classificação ou agregação. Esta limitação é imposta para garantir que os dados possam ser divididos em um ambiente Hadoop e durante a mineração dentro da base de dados. Esta limitação não se aplica se os scripts para pontuação do modelo forem executados em Servidor do SPSS Modeler. Nós de Extensão de Extensão e Nós Modelo de Extensão não tem essa limitação.

- **Converter campos de sinalização.** Especifica como os campos de sinalização são tratados. Há duas opções: **Sequências para fator, Números inteiros e reais para dobro** e **Valores lógicos (True, False)**. Se você selecionar **Valores lógicos (True, False)** os valores originais dos campos de sinalização serão perdidos. Por exemplo, se um campo tiver os valores Masculino e Feminino, eles serão alterados para True e False.
- **Converter os valores ausentes para o valor R 'não disponível' (NA).** Quando selecionado, quaisquer valores ausentes são convertidos para o valor NA R. O valor NA é usado por R para identificar valores ausentes. Algumas funções R que você usa podem ter um argumento que pode ser usado para controlar como a função se comporta quando os dados contêm NA. Por exemplo, a função pode permitir que você opte por excluir automaticamente os registros que contêm NA. Se essa opção não for selecionada, qualquer valor omissos será passado para R inalterado e poderá causar erros quando o script R for executado.
- **Converter campos de data/hora em classes de R com controle especial para fusos horários** Quando selecionado, as variáveis com os formatos de hora ou data/hora são convertidos em objetos de data/hora de R. Você deve selecionar uma das seguintes opções:
 - **Variáveis POSIXct.** R com formatos de data ou data e hora são convertidas em objetos POSIXct R.
 - **R POSIXlt (lista).** Variáveis com formatos de data ou data e hora são convertidas em objetos POSIXlt R.

Nota: Os formatos POSIX são opções avançadas. Use essas opções somente se o script R especificar que os campos data/hora sejam tratados de uma forma que requeira esses formatos. Os formatos POSIX não se aplicam às variáveis com formatos de hora.

As opções que são selecionadas para os **Convert flag fields**, **Converte valores ausentes para o valor de R 'not available' (NA)**, e **Convert data / time campos para classes R com controle especial para fusos horários** controles não são reconhecidos quando o nugget modelo de Extensão é executado contra um banco de dados. Em vez disso, quando o nó é executado com relação a um banco de dados, os valores padrão para esses controles são usados:

- **Converter campos de sinalização** é configurado como **Sequências para fator, Números inteiros e Reais para dobro**.
- **Converter valores omissos no valor de R 'não disponível' (NA)** está selecionado.
- **Converter campos de data/hora em classes de R com controle especial para fusos horários** não está selecionado.

Modelo de extensão nugget-Graph Output

A guia **Graph Output** está presente no nugget de modelo de Extensão se solicitado, selecionando a caixa de seleção **Exibir gráficos R como HTML** na guia **Opções do Modelo** da caixa de diálogo do nó da Extensão Modelo. Os gráficos resultantes da execução do script R de construção de modelo podem ser exibidos nessa guia. Por exemplo, se o seu script R contiver uma chamada para a função R `plot`, o gráfico resultante será exibido nesta guia. Se você executar o script de construção de modelo novamente, sem ter especificado primeiro um nome diferente para o modelo, o conteúdo da guia **Saída de Gráfico** da execução anterior será sobreescrito.

Nugget modelo de extensão-Guia de saída de texto

A guia **Saída de Texto** está presente no nugget de modelo de Extensão se solicitado selecionando a caixa de seleção **Exibir saída de texto do R R** na guia **Opções do Modelo** da caixa de diálogo do nó do Modelo de Extensão. Essa guia pode exibir somente saída de texto. Qualquer saída de texto que seja produzida executando seu script de Modelo de Extensão é exibida nesta guia. Se você executar o script de Modelo de Extensão novamente, sem ter especificado primeiro um nome diferente para o modelo, o conteúdo da guia **Saída de Texto** da execução anterior será sobreescrito. A saída de texto não pode ser editada.

Nota:

- Se você incluir uma chamada para a função `sink` de R em seu script, qualquer saída produzida após essa função será salva no arquivo especificado e não será exibida na guia **Saída de Texto**.
- Mensagens de erro ou avisos que resultam da execução do seu script de Modelo de Extensão são sempre exibidos na guia **Saída do Console** do nó do Modelo de Extensão.

Nugget modelo de extensão-Guia de Saída do Console

A guia **Saída do Console** está sempre presente no nugget modelo de Extensão. Ele contém qualquer saída que é recebida quando o script R ou Python para script Spark na guia Syntax executa (por exemplo, se usando um script R, ele mostra saída recebida do console R quando o script R no campo **R model scoring sintaxe** na guia **Syntax** do nugget modelo de Extensão é executado). Esta saída inclui quaisquer mensagens de erro R ou Python ou avisos que são produzidos quando o script R ou Python é executado, e qualquer saída de texto do console R. A saída pode ser usada, principalmente, para depurar o script.

Toda vez que o script de pontuação do modelo é executado, o conteúdo da guia **Output Console** é sobreescrito com a saída recebida do console R ou do Python para Spark. A saída de console não pode ser editada.

Nó de transformação de extensão

Com o nó Extensão Transforme, é possível tirar dados de um fluxo IBM SPSS Modeler e aplicar transformações nos dados usando o R scripting ou Python para scripting Spark. Quando os dados tiverem sido modificados, eles serão retornados ao fluxo para maior processamento, construção de modelo e escoragem de modelo. O Extensão Transformar o nó possibilita a transformação de dados usando algoritmos gravados em R ou Python para Spark, e possibilita que o usuário desenvolva métodos de transformação de dados adaptados a um determinado problema.

Para usar este nó com R, você deve instalar IBM SPSS Modeler - Essentials for R. Consulte o *IBM SPSS Modeler -Essentials for R: instruções de instalação* para obter instruções de instalação e informações de compatibilidade. Deve-se também ter uma versão compatível de R instalada no computador.

Nó Transformar nó-Guia Sintaxe

Selecione seu tipo de sintaxe - **R** ou **Python for Spark**. Consulte as informações a seguir para obter mais informações. Quando sua sintaxe estiver pronta, você pode clicar em **Executar** para executar o nó Extensão Transforme.

Sintaxe de R

R Sintaxe. É possível inserir, ou colar, a sintaxe customizada do script R para a análise de dados nesse campo.

Converter campos de sinalização. Especifica como os campos de sinalização são tratados. Há duas opções: **Sequências para fator, Números inteiros e reais para dobro** e **Valores lógicos (True, False)**. Se você selecionar **Valores lógicos (True, False)** os valores originais dos campos de sinalização serão perdidos. Por exemplo, se um campo tiver os valores Masculino e Feminino, eles serão alterados para True e False.

Converter os valores ausentes para o valor R 'não disponível' (NA). Quando selecionado, quaisquer valores ausentes são convertidos para o valor NA R. O valor NA é usado por R para identificar valores ausentes. Algumas funções R que você usa podem ter um argumento que pode ser usado para controlar como a função se comporta quando os dados contêm NA. Por exemplo, a função pode permitir que você opte por excluir automaticamente os registros que contêm NA. Se essa opção não for selecionada, qualquer valor omissos será passado para R inalterado e poderá causar erros quando o script R for executado.

Converter os campos data/hora em classes de R com controle especial para fusos horários. Quando selecionado, as variáveis com formatos de data ou data/hora são convertidas em objetos de data/hora de R. Você deve selecionar uma das seguintes opções:

- **Variáveis POSIXct.** R com formatos de data ou data e hora são convertidas em objetos POSIXct R.
- **R POSIXlt (lista).** Variáveis com formatos de data ou data e hora são convertidas em objetos POSIXlt R.

Nota: Os formatos POSIX são opções avançadas. Use essas opções somente se o script R especificar que os campos data/hora sejam tratados de uma forma que requeira esses formatos. Os formatos POSIX não se aplicam a variáveis com formatos de tempo.

Sintaxe do Python

Python Sintaxe. Você pode inserir, ou colar, personalizada sintaxe de script Python para análise de dados neste campo. Para obter mais informações sobre o Python para Spark, consulte [Python para Spark](#) e [Scripting com Python para Spark](#).

Nó Transformar nó-Guia de saída do console

A guia **Saída do Console** contém qualquer saída que é recebida quando o script R ou Python para script Spark na guia Syntax executa (por exemplo, se usando um script R, ele mostra saída recebida do console R quando o script R no campo **R Syntax** na guia **Syntax** é executado). Essa saída pode incluir mensagens de erro R ou Python ou avisos que são produzidos quando o script R ou Python é executado. A saída pode ser usada, principalmente, para depurar o script. A guia **Saída de console** também contém o script a partir do campo **R Sintaxe** ou **Sintaxe Python**.

Toda vez que o script Transformar de Extensão é executado, o conteúdo da guia **Output Console** é sobrescrito com a saída recebida do console R ou Python para Spark. A saída não pode ser editada.

Nó de importação de extensão

Com o nó de Importação de Extensão, é possível executar o R ou Python para scripts de Spark para importar dados.

Nó de Importação de Extensão-Guia Sintaxe

Selecione seu tipo de sintaxe - **R** ou **Python for Spark**. Em seguida, insira ou cole seu script customizado para a importação de dados. Quando a sua sintaxe estiver pronta, você pode clicar em **Executar** para executar o nó de Importação de Extensão.

Exemplo de R

```
# import R demo data cars to modeler
modelerData <- cars

# write the data model that matches the data
var1<-c(fieldName="speed",fieldLabel="",fieldStorage="integer",fieldMeasure="",fieldFormat="",
fieldRole="")
var2<-c(fieldName="dist",fieldLabel="",fieldStorage="integer",fieldMeasure="",fieldFormat="",
fieldRole="")
modelerDataModel<-data.frame(var1, var2)
```

Exemplo de Python for Spark

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

ctxt = spss.pyspark.runtime.getContext()
if ctxt.isComputeDataModelOnly():
    _schema = StructType([StructField("Age", LongType(), nullable=True), \
        StructField("Sex", StringType(), nullable=True), \
        StructField("BP", StringType(), nullable=True), \
        StructField("Cholesterol", StringType(), nullable=True), \
        StructField("Na", DoubleType(), nullable=True), \
        StructField("K", DoubleType(), nullable=True), \
        StructField("Drug", StringType(), nullable=True)])
    ctxt.setSparkOutputSchema(_schema)
else:
    sqlContext = ctxt.getSparkSQLContext()
    # the demo data is in modeler installation path
    df = sqlContext.read.option("inferSchema", "true").option("header", "true").csv("/opt/IBM/
SPSS/ModelerServer/Cloud/demos/DRUG1n")
    ctxt.setSparkOutputData(df)
    df.show()
    # print (df.dtypes[:])
```

Nó de Importação de Extensão-Guia de Saída

A guia **Saída do Console** contém qualquer saída que é recebida quando o script R ou Python para script Spark na guia Syntax executa (por exemplo, se usando um script R, ele mostra saída recebida do console R quando o script R no campo **R Syntax** na guia **Syntax** é executado). Essa saída pode incluir mensagens de erro R ou Python ou avisos que são produzidos quando o script R ou Python é executado. A saída pode ser usada, principalmente, para depurar o script. A guia **Saída de console** também contém o script a partir do campo **R Sintaxe** ou **Sintaxe Python**.

Toda vez que o script de Importação de Extensão é executado, o conteúdo da guia **Saída do Console** é sobreescrito com a saída recebida do console R ou Python para Spark. A saída não pode ser editada.

Filtrando ou renomeando campos

É possível renomear ou excluir campos em qualquer ponto em um fluxo. Por exemplo, como pesquisador médico, você pode não estar preocupado com o nível de potássio (dados em nível de campo) dos pacientes (dados em nível de registro); portanto, é possível filtrar o campo K (potássio). Isso pode ser

feito utilizando um nó Filtro separado ou utilizando a guia Filtro em um nó de origem ou de saída. A funcionalidade é a mesma, independentemente de qual nó ele estiver sendo acessado.

- A partir de nós de origem, como Arquivo Variável, Arquivo Fixo, Estatísticas Arquivo, XML ou Importação de Extensão, você pode renomear ou filtrar campos conforme os dados são lidos em IBM SPSS Modeler.
- Utilizando um nó Filtro, é possível renomear ou filtrar os campos em qualquer ponto no fluxo.
- Nos nós Exportação do Estatísticas, Transformação do Estatísticas, Modelo do Estatísticas e Saída do Estatísticas, é possível filtrar ou renomear campos para conformidade com os padrões de nomenclatura do IBM SPSS Estatísticas.
- É possível utilizar a guia Filtro em qualquer um dos nós acima para definir ou editar conjuntos de múltiplas respostas.
- Por fim, é possível utilizar um nó Filtro para mapear os campos de um nó de origem para outro.

Capítulo 2. Extensões

Extensões são componentes personalizados que estendem as capacidades do IBM SPSS Modeler. As extensões são empacotadas em pacotes de extensão (.mpe arquivos) e são instaladas para IBM SPSS Modeler. As extensões podem ser criadas por qualquer usuário e compartilhadas com outros usuários ao compartilhar o pacote configurável de extensão associado.

Os utilitários a seguir são fornecidos para trabalhar com extensões

- O “Hub de extensão” na página 27, que é acessado a partir de **Extensões > Hub de Extensão**, é uma interface para procurar, fazer download e instalar extensões da coleção do Análise preditivaIBM SPSS em GitHub. No diálogo Hub de Extensão, também é possível visualizar detalhes das extensões que estiverem instaladas no seu computador, obter atualizações para extensões instaladas e remover extensões.
- É possível instalar o pacote configurável de extensão que estiver armazenado no seu computador local a partir de **Extensões > Instalar Pacote Configurável de Extensão Local**.
- É possível usar o Construtor de Diálogo Personalizado para Extensões para criar uma extensão que inclua uma interface com o usuário, que é referida como um diálogo de nó customizado. Diálogos de nó customizado geram script R ou Python para script Spark que realiza tarefas associadas à extensão. Você projeta o script gerado como parte do projeto do diálogo customizado.

Hub de extensão

No diálogo Hub de Extensão, é possível executar as seguintes tarefas:

- Explore extensões disponíveis na coleta do Análise preditivaIBM SPSS no GitHub. É possível selecionar extensões para instalar agora ou fazer download de extensões selecionadas e instalá-las mais tarde.
- Obter versões atualizadas de extensões que já estiverem instaladas no seu computador.
- Obter detalhes sobre as extensões que já estiverem instaladas no seu computador.
- Remover extensões que já estiverem instaladas no seu computador.

Para fazer download ou remover extensões:

1. Nos menus, escolha: **Extensões > Hub de Extensão**
2. Selecione as extensões que deseja fazer download ou remover e clique em **OK**. Todas as seleções que forem feitas nas guias Explorar e Instaladas são processadas ao clicar em **OK**.

Por padrão, as extensões que são selecionadas para download são transferidas por download e instaladas no seu computador. Na guia Configurações, é possível escolher fazer download das extensões selecionadas em uma localização especificada sem instalá-las. Em seguida, é possível instalá-las posteriormente ao escolher **Extensões > Instalar Pacote Configurável de Extensão Local**. Para Windows, é possível instalar uma extensão clicando duas vezes no arquivo de pacote configurável de extensão.

Importante:

- Para Windows 7 e later, instalar uma versão atualizada de um pacote configurável de extensão existente pode requerer a execução do IBM SPSS Modeler com privilégios de administrador. É possível iniciar o IBM SPSS Modeler com privilégios de administrador ao clicar com o botão direito no ícone para o IBM SPSS Modeler e escolher **Executar como administrador**. Em particular, se você receber uma mensagem de erro indicando que um ou mais pacotes configuráveis de extensão não puderam ser instalados, tente executá-los com privilégios de administrador.
- Se estiver se conectando à Internet por meio de um proxy, você poderá receber um erro como "Alguns recursos não estão disponíveis porque nenhuma conexão de Internet foi detectada" quando tentar abrir o Hub de extensão por meio da opção de menu **Extensões > Hub de extensão**. Para resolver isso,

você inclui os seguintes parâmetros para as opções # JVM no arquivo jvm.cfg (localizado no diretório config da sua instalação ModeladorSPSS). Salve o arquivo e reinicie o ModeladorSPSS.

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

Nota: A licença com a qual você concorda quando instalar uma extensão pode ser visualizada em qualquer momento posterior clicando em **Mais info ...** para a extensão na guia Instalada.

Guia Explorar

A guia Explorar exibe todas as extensões disponíveis da coleção Análise preditivaIBM SPSS no GitHub (<https://ibmpredictiveanalytics.github.io/>). A partir da guia Explorar, é possível selecionar novas extensões para fazer o download e instalar, bem como selecionar atualizações para extensões já instaladas em seu computador. A guia Explorar requer uma conexão de Internet.

- Para cada extensão, o número da versão mais recente e a data associada dessa versão são exibidos. Uma sumarização breve da extensão também é fornecida. Para extensões que já estiverem instaladas no seu computador, o número da versão instalada também é exibido.
- É possível visualizar informações detalhadas sobre a extensão clicando em **Mais informações**. Quando uma atualização está disponível, **Mais informações** exibe informações sobre a atualização.
- É possível visualizar os pré-requisitos de execução de uma extensão, como se o IBM SPSS Modeler - Plug-in de integração para R é necessário ao clicar em **Pré-requisitos**. Quando uma atualização está disponível, **Pré-requisitos** exibe informações sobre a atualização.

Refinar por

É possível refinar o conjunto de extensões que são exibidas. É possível refinar por categorias gerais de extensões, pelo idioma no qual a extensão é implementada, pelo tipo de organização que forneceu a extensão ou pelo estado da extensão. Para cada grupo, como Categoria, é possível selecionar diversos itens pelos quais refinar a lista de extensões exibidas. Também é possível refinar por termos de procura. As proezas não fazem distinção entre maiúsculas e minúsculas, e o asterisco (*) é tratado como qualquer outro caractere e não indica uma proezza curinga.

- Para refinar a lista de extensões exibidas, clique em **Aplicar**. Pressionar a tecla Enter quando o cursor estiver na caixa **Procurar** possui o mesmo efeito que clicar em **Aplicar**.
- Para reconfigurar a lista para exibir todas as extensões disponíveis, exclua qualquer texto na caixa **Procurar**, desmarque todos os itens e clique em **Aplicar**.

Como obter plug-ins de integração

Para obter o IBM SPSS Modeler - Plug-in de integração para R:

Instale IBM SPSS Modeler - Essentials for R, disponível a partir de https://github.com/IBMPredictiveAnalytics/R_Essentials_Modeler/releases/ ou a comunidade IBM ' SPSS Estatísticas em <https://www.ibm.com/products/spss-statistics/support>. IBM SPSS Modeler - Essentials for R inclui o IBM SPSS Modeler - Plug-in de integração para R. Fundamentos para R não inclui a linguagem de programação R. Antes de instalar IBM SPSS Modeler - Essentials for R você precisará instalar a versão R 4.0 se ele ainda não estiver instalado. Ele está disponível no <https://cran.r-project.org/>. Recomenda-se baixar e instalar o R 4.0.x.

Nota: Se você está instalando Fundamentos para R em um computador que não tem acesso à internet e você planeja usar os comandos de extensão do R scripts que estão incluídos com Fundamentos para R, então você deve obter quaisquer pacotes R que são necessários por aqueles scripts e instalá-los manualmente em R. Para determinar quais pacotes R são necessários para um comando de extensão de R script, abra o diálogo de Extensão Hub (**Extensões > Hub de Extensão**), clique na guia **Instalado**, e em seguida clique em **Mais informações** para a extensão desejada. Os pacotes R necessários estão listados no diálogo Detalhes da Extensão. Os pacotes r podem ser obtidos em qualquer um dos sites de espelhos R CRAN, que são acessados a partir de <http://www.r-project.org/>. Certifique-se de obter as versões dos

pacotes que combinam com a sua versão R. Os pacotes específicos da versão estão disponíveis a partir de links na página "Pacotes Contribuidos" do site de espelho CRAN.

Guia Instaladas

A guia Instaladas exibe todas as extensões instaladas em seu computador. Na guia Instaladas, é possível selecionar atualizações para as extensões instaladas disponíveis na coleção do Análise preditivaIBM SPSS em GitHub e é possível remover extensões. Para obter atualizações das extensões instaladas, é necessário ter uma conexão de Internet.

- Para cada extensão, o número da versão instalada é exibido. Quando uma conexão de Internet está disponível, o número da versão mais recente e a data associada dessa versão são exibidos. Uma sumarização breve da extensão também é fornecida.
- É possível visualizar informações detalhadas sobre a extensão clicando em **Mais informações**. Quando uma atualização está disponível, **Mais informações** exibe informações sobre a atualização.
- É possível visualizar os pré-requisitos de execução de uma extensão, como se o IBM SPSS Modeler - Plug-in de integração para R é necessário ao clicar em **Pré-requisitos**. Quando uma atualização está disponível, **Pré-requisitos** exibe informações sobre a atualização.

Refinar por

É possível refinar o conjunto de extensões que são exibidas. É possível refinar por categorias gerais de extensões, pelo idioma no qual a extensão é implementada, pelo tipo de organização que forneceu a extensão ou pelo estado da extensão. Para cada grupo, como Categoria, é possível selecionar diversos itens pelos quais refinar a lista de extensões exibidas. Também é possível refinar por termos de procura. As proezas não fazem distinção entre maiúsculas e minúsculas, e o asterisco (*) é tratado como qualquer outro caractere e não indica uma proezza curinga.

- Para refinar a lista de extensões exibidas, clique em **Aplicar**. Pressionar a tecla Enter quando o cursor estiver na caixa **Procurar** possui o mesmo efeito que clicar em **Aplicar**.
- Para reconfigurar a lista para exibir todas as extensões disponíveis, exclua qualquer texto na caixa **Procurar**, desmarque todos os itens e clique em **Aplicar**.

Extensão privada

Extensões privadas são extensões instaladas em seu computador, mas não estão disponíveis na coleção do Análise preditivaIBM SPSS em GitHub. Os recursos para refinar o conjunto de extensões exibidas e para visualizar os pré-requisitos para executar uma extensão não estão disponíveis para extensões privadas.

Nota: Ao usar o Hub de Extensão sem uma conexão de Internet, alguns dos recursos da guia Instaladas podem não estar disponíveis.

Configurações

A guia Configurações especifica se as extensões que são selecionadas para download são transferidas por download e, em seguida, instaladas ou se são transferidas por download, mas não instaladas. Essa configuração se aplica a novas extensões e atualizações para extensões existentes. É possível optar por fazer download das extensões sem instalá-las se estiver fazendo download das extensões para distribuir para outros usuários em sua organização. Também é possível optar por escolher fazer download, mas não instalar, as extensões se você não tiver os pré-requisitos para executar as extensões, mas planeja obter os pré-requisitos.

Se escolher fazer download das extensões sem instalá-las, será possível instalá-las posteriormente ao escolher **Extensões > Instalar Pacote Configurável de Extensão Local**. Para Windows, é possível instalar uma extensão clicando duas vezes no arquivo de pacote configurável de extensão.

Detalhes de Extensão

A caixa de diálogo Detalhes de Extensão exibe as informações que foram fornecidas pelo autor da extensão. Além de informações necessárias, como Resumo e Versão, o autor pode ter incluído URLs para locais de relevância, como a página inicial do autor. Se a extensão foi transferida por download do Hub de Extensão, então uma licença pode ser visualizada ao clicar em **Visualizar licença**.

Nós Customizados. A tabela Nós Customizados lista os diálogos de nó customizado incluídos na extensão.

Nota: Instalar uma extensão que contenha um diálogo de nó customizado pode requerer a reinicialização do IBM SPSS Modeler para ver a entrada para o diálogo de nó na tabela Nós Customizados.

Dependências. O grupo Dependências lista complementos que são necessários para executar os componentes incluídos na extensão.

- **Plug-In de integração para R.** Os componentes para uma extensão podem necessitar do Plug-in de integração para R.
- **Pacotes R.** Lista quaisquer pacotes R que forem necessários pela extensão. Consulte o tópico “[Pacotes R necessários](#)” na página 31 para obter mais informações.

Instalando pacotes configuráveis de extensão

Para instalar um pacote configurável de extensão que estiver armazenado no seu computador local:

1. No menu, escolha:

Extensões > Instalar pacote configurável de extensão local...

2. Selecione um pacote configurável de extensão. Os bundles de extensão possuem um tipo de arquivo de mpe.

Importante: Para usuários de Windows 7 e versões mais recentes de Windows, instalar uma versão atualizada de um pacote configurável de extensão pode requerer a execução do IBM SPSS Modeler com privilégios de administrador. É possível iniciar o IBM SPSS Modeler com privilégios de administrador ao clicar com o botão direito no ícone para o IBM SPSS Modeler e escolher **Executar como administrador**. Em particular, se você receber uma mensagem de erro indicando que um ou mais pacotes configuráveis de extensão não puderam ser instalados, tente executá-los com privilégios de administrador.

Locais de instalação para extensões

Por padrão, extensões são instaladas em um local geral que pode ser gravado pelo usuário para seus sistemas operacionais.

Você pode substituir o local padrão, definindo um caminho com a variável de ambiente `IBM_SPSS_MODELER_EXTENSION_PATH`. O local especificado deve existir no computador de destino. Depois de configurar `IBM_SPSS_MODELER_EXTENSION_PATH`, você deve reiniciar IBM SPSS Modeler para que as alterações surtam efeito.

Para criar uma variável de ambiente em Windows, No Painel de Controle:

Windows 7

1. Selecione contas do usuário.
2. Clique em **Alterar minhas variáveis de ambiente**.
3. Clique em **Novo**, digite o nome da variável de ambiente (por exemplo, `IBM_SPSS_MODELER_EXTENSION_PATH`) no campo **Nome da Variável** e digite o caminho ou os caminhos no campo de valor Variable.

Windows 8 ou posterior

1. Selecione Sistema.

2. Selecione a guia Avançado e clique em **Variáveis de ambiente**. A guia Avançado é acessada nas configurações avançadas do sistema.
3. Na seção Variáveis do Usuário, clique em **Novo**, digite o nome da variável de ambiente (por exemplo, IBM_SPSS_MODELER_EXTENSION_PATH) no campo **Nome da Variável** e digite o caminho ou os caminhos no campo **Valor Variável**.

Importante: Para usuários de Windows 7 e versões mais recentes de Windows, instalar uma versão atualizada de um pacote configurável de extensão pode requerer a execução do IBM SPSS Modeler com privilégios de administrador. É possível iniciar o IBM SPSS Modeler com privilégios de administrador ao clicar com o botão direito no ícone para o IBM SPSS Modeler e escolher **Executar como administrador**. Em particular, se você receber uma mensagem de erro indicando que um ou mais pacotes configuráveis de extensão não puderam ser instalados, tente executá-los com privilégios de administrador.

Executando fluxos no modo batch que contêm extensões

Para usar extensões do Custom Dialog Builder com o Modeler Batch:

1. No Modeler Client, use o Custom Dialog Builder para criar e instalar o seu nó personalizado. Os arquivos do nó de extensão estarão no diretório a seguir, por padrão:
 - **Windows:** C:\ProgramData\IBM\SPSS\Modeler\<version>\CDB\
 - **Mac:** /Users/yourname/Library/Application Support/IBM/SPSS/Modeler/\<version>/CDB/
2. Copie o diretório CDB para a máquina do Batch do Modelador.
3. Exporte a variável de ambiente IBM_SPSS_MODELER_EXTENSION_PATH para o novo diretório. Agora é possível executar o fluxo no modo batch.

Pacotes R necessários

Se não tiver acesso à Internet, você precisará obter qualquer pacote R requerido para uma extensão em particular que não seja localizado em seu computador com quem tiver. É possível visualizar a lista de pacotes R requeridos na caixa de diálogo Detalhes da extensão quando a extensão estiver instalada. Consulte o tópico “Detalhes de Extensão” na página 30 para obter mais informações. Os pacotes podem ser transferidos por download de <http://www.r-project.org/> e, em seguida, instalados de dentro de R. Para obter detalhes, consulte o guia de *Instalação e administração do R*, distribuído com o R.

Nota: Para usuários de UNIX (incluindo Linux), pacotes são transferidos por download em forma de origem e, em seguida, compilados. Isso requer que as ferramentas apropriadas estejam instaladas na sua máquina. Consulte o guia *Instalação e Administração do R* para obter detalhes. Em particular, usuários Debian devem instalar o pacote r-base-dev a partir de apt-get install r-base-dev.

Criando e gerenciando nós customizados

O Construtor de diálogo personalizado para Extensões cria sintaxe de comando para serem usados dentro de fluxos ModeladorSPSS.

Usando Construtor de Diálogo Personalizado para Extensões é possível:

- Criar um diálogo de nó customizado para executar um nó que seja implementado no R ou no Apache Spark (via Python). Veja “Construindo o modelo de script” na página 34 para obter mais informações.
- Abra um arquivo contendo a especificação de um diálogo customizado - talvez criado por outro usuário - e inclua o diálogo na sua instalação do IBM SPSS Modeler, fazendo, opcionalmente, suas próprias modificações.
- Salve a especificação de um diálogo customizado para que outros usuários possam incluí-la em suas instalações do IBM SPSS Modeler.
- Crie nós customizados e grave scripts Python for Spark para ler dados de onde quer que sua origem de dados está e grave dados em qualquer formato de dados suportado pelo Apache Spark.

Veja “[Importando e exportando dados usando Python para Spark](#)” na página 60 para obter mais informações.

- Crie nós customizados e grave scripts R para ler dados de onde quer que sua origem de dados está e grave dados em qualquer formato de dados suportado pelo R. Veja “[Importando e exportando dados usando R](#)” na página 61 para obter mais informações.

No Construtor de Diálogo Personalizado para Extensões, você cria ou modifica diálogos de nó customizado em extensões. Quando você abre o Construtor de Diálogo Personalizado para Extensões, uma nova extensão que contém um diálogo de nó customizado vazio é criado. Quando você salva ou instala diálogos de nó customizado a partir do Construtor de Diálogo Personalizado para Extensões, eles são salvos ou instalados como parte de uma extensão.

Nota:

- Não é possível criar sua própria versão de um novo diálogo de nó para um nó IBM SPSS Modeler padrão.
- O script não é suportado para nós que são criados com o Construtor de diálogo personalizado, incluindo nós R do Construtor de diálogo personalizado e nós Python do Construtor de diálogo personalizado.

Como iniciar o Construtor de Diálogo Personalizado para Extensões

Nos menus, escolha **Extensões > Construtor de Diálogo de Nó Customizado**

Nota:

- Os nós Python dependem do ambiente Spark.
- Os scripts Python devem usar a API do Spark porque os dados serão apresentados na forma de um DataFrame.
- Os nós antigos criados na versão 17.1 ainda serão executados com relação a Servidor analíticoIBM SPSS (os dados são originados de um nó de origem Servidor analíticoIBM SPSS e não foram extraídos para o IBM SPSS Modeler Server). Os novos nós do Python e do Construtor de Diálogo Customizado criados na versão 18.0 ou posterior podem ser executados no IBM SPSS Modeler Server.
- Ao instalar o Python, certifique-se de que todos os usuários tenham permissão para acessar a instalação do Python.
- Se você desejar usar a Biblioteca do Machine Learning (MLlib), deverá instalar uma versão do Python que inclua NumPy. Em seguida, deve-se configurar o IBM SPSS Modeler Server (ou o servidor local no IBM SPSS Modeler Client) para usar sua instalação do Python . Para obter detalhes, consulte “[Script com Python for Spark](#)” na página 2.

Layout do construtor de diálogo personalizado

Tela de Diálogo

A tela de diálogo é a área do Construtor de diálogo personalizado em que o layout de seu diálogo nó é designado.

Área de janela Propriedades

A área de janela de propriedades é a área do Construtor de diálogo personalizado na qual as propriedades dos controles que criam o diálogo node são especificadas, bem como as propriedades do próprio diálogo, como a tipo de nó.

Paleta de ferramentas

A paleta de ferramentas fornece o conjunto de controles que pode ser incluído em um diálogo nó customizado. É possível mostrar ou ocultar a Paleta de Ferramentas, escolhendo Paleta de Ferramentas no menu **Visualizar**.

Modelo de de Sintaxe

O Modelo de de Sintaxe especifica a R script or Python for Spark script que é gerada pelo diálogo nó customizado. É possível mover a área de janela Modelo de de Sintaxe para uma janela, clicando e **Mover para nova janela**. Para mover uma janela Modelo de de Sintaxe de volta para o Construtor de diálogo personalizado, clique em **Restaurar para a janela principal**.

Construindo um diálogo de nó customizado

As etapas básicas envolvidas na construção de um diálogo de nó customizado são:

1. Especifique as propriedades do diálogo de nó dialog em si, como o título que aparece quando o diálogo de nó é ativado e o local do novo nó dentro das paletas IBM SPSS Modeler. Consulte o tópico “Propriedades do diálogo” na página 33 para obter mais informações.
2. Especifique os controles, como seletores de campo e caixas de seleção, que criam o diálogo de nó e quaisquer subdiálogos. Consulte o tópico “Tipos de controle” na página 36 para obter mais informações.
3. Crie o modelo de script que especifica o código R ou o código Python for Spark que é gerado pelo diálogo de nó. Consulte o tópico “Construindo o modelo de script” na página 34 para obter informações adicionais.
4. Especifique propriedades da extensão que contenham seu diálogo de nó. Veja o tópico “Propriedades de Extensão” na página 54 para obter mais informações.
5. Instale a extensão que contém o diálogo node para IBM SPSS Modeler e / ou salvar a extensão em um arquivo de extensão (.mpe). Consulte o tópico “Gerenciando diálogos nó customizado” na página 57 para obter mais informações.

É possível visualizar seu diálogo de nó conforme ele é construído. Consulte o tópico “Visualizando um diálogo de nó customizado” na página 36 para obter mais informações.

Propriedades do diálogo

A janela Construtor de diálogo personalizado mostra as propriedades para o diálogo de nó e para o controle de interface com o usuário selecionado. Para visualizar e configurar as Propriedades de Diálogo, clique na tela em uma área fora de qualquer controle. Sem controles na tela, as Propriedades de Diálogo estarão sempre visíveis.

Nome de diálogo. A propriedade Nome de Diálogo é requerida e especifica um nome exclusivo com o qual associar o diálogo de nó . Para minimizar a possibilidade de conflitos de nome, você pode desejar prefixar o nome de com um identificador para sua organização, como uma URL.

Título. A propriedade Título especifica o texto a ser exibido na barra de título da caixa de diálogo de nó .

Arquivo de ajuda. A propriedade Arquivo de ajuda é opcional e especifica o caminho para um arquivo de ajuda para o diálogo de nó . Este é o arquivo que será ativado quando o usuário clicar no botão **Ajuda** no diálogo. Arquivos de ajuda devem estar no formato HTML. Uma cópia do arquivo de ajuda especificado é incluída com as especificações para o diálogo de nó quando o diálogo de nó é instalado ou salvo. O botão Ajuda no diálogo de tempo de execução é oculto se não houver arquivo de ajuda associado.

- Versões localizadas do arquivo de ajuda que existem no mesmo diretório que o arquivo de ajuda são automaticamente incluídas no diálogo de nó quando você inclui o arquivo de ajuda. As versões localizadas do arquivo de ajuda são denominadas '`<Help File>_<language identifier>.htm`'. Para obter mais informações, consulte o tópico “Criando versões localizadas dos diálogos de nó customizados” na página 59.
- Arquivos de apoio, como arquivos de imagem e folhas de estilo, podem ser incluídos no diálogo de nó primeiro salvando o diálogo de nó. Você, então, inclui manualmente os arquivos de apoio no arquivo do diálogo de nó (.cfe). Para obter informações sobre como acessar e modificar manualmente arquivos do diálogo de nó, consulte a seção intitulada "Para localizar sequências de diálogo" no tópico “Criando versões localizadas dos diálogos de nó customizados” na página 59.

Tipo de Script. Especifica o tipo de script que pode ser usado para construir o Modelo de Script. No IBM SPSS Modeler, o script R ou o script Python for Spark poderão ser usados.

Pontuação do modelo. Especifica se o modelo que é construído usando o script de construção de modelo é usado para pontuação.

Tipo de Nô. Especifica o tipo de nó que será criado ao instalar seu diálogo de nó.

Paleta. Especifica a paleta para a qual o nó recém-criado será incluído quando você instalar o diálogo de nó.

Ícone de nó. Clique no botão de reticências (...) para selecionar uma imagem a ser usada como o ícone de nó para o nó recém-criado. A imagem que você escolher deve ser um arquivo .gif.

Efetuando layout em controle na tela de diálogo

É possível incluir controles a um diálogo de nó customizado arrastando-os da paleta de ferramentas para a tela de diálogo. Para garantir a consistência com diálogos de nós integrados, a tela do diálogo deve ser dividida em quatro colunas funcionais em que seja possível colocar os controles.

- A primeira coluna (mais à esquerda) destina-se principalmente a um controle Seletor de campo.
- Os botões do subdiálogo devem estar na coluna mais à direita (por exemplo, a terceira coluna se apenas três colunas forem usadas) e nenhum outro controle pode estar na mesma coluna como botões de subdiálogo. A este respeito, a quarta coluna pode conter apenas botões de subdiálogos.

Embora não seja mostrado na tela de diálogo, quando o diálogo de nó é instalado no IBM SPSS Modeler, os botões apropriados são incluídos ao diálogo (por exemplo: **OK**, **Cancelar**, **Aplicar**, **Reconfigurar** e, se apropriado, **Ajuda** e **Executar**). A presença e as localizações desses botões são automáticas. No entanto, o botão **Ajuda** é oculto se não houver arquivo de ajuda associado ao diálogo de nó (conforme especificado pela propriedade Arquivo de Ajuda em Propriedades do Diálogo).

É possível alterar a ordem vertical dos controles dentro de uma coluna, arrastando-os para cima ou para baixo, mas a exata posição dos controles é determinada automaticamente por você. No tempo de execução, os controles são redimensionados de forma apropriada quando o próprio diálogo for redimensionado. Controles como seletores de campo são expandidos automaticamente para preencher o espaço disponível abaixo deles.

Construindo o modelo de script

O modelo de script especifica o script R ou o script Python para Spark que o diálogo de nó personalizado irá gerar. Um diálogo de nó customizado único pode ser usado para especificar uma ou mais operações que executarão em sequência.

O modelo de script deve ser composto de *texto estático*. O texto estático é diferente para o controle de texto estático; é o código R ou Python para código Spark que é sempre gerado quando o nó é executado. Por exemplo, os nomes de comando e especificações de subcomando que não dependem da entrada do usuário são textos estáticos. O modelo de script também pode ser composto de identificadores de controle que são substituídos no tempo de execução por valores dos controles associados do diálogo customizado do nó. Por exemplo, o conjunto de campos especificados em um seletor de campo é representado com o identificador de controle para o controle de seletor de campo.

Para construir o modelo de script

1. Para o texto estático que não depende de valores especificados pelo usuário, digite o script R ou o script Python para Spark como você faria em, por exemplo, o campo **sintaxe de construção do modelo R** do nó do R Build.
2. Adiciona identificadores de controle do formulário %%Identifier%% nos locais em que deseja inserir script R ou Python para script Spark gerado por controles, onde Identifier é o valor da propriedade Identifier para o controle.
 - Você pode inserir um identificador de controle, selecionando uma linha na tabela de identificadores, clicando com o botão direito do mouse e selecionando **Adicionar ao modelo de script**. Você

também pode inserir um identificador de controle clicando com o botão direito do mouse sobre um controle na tela e selecionando **Adicionar ao modelo de script**.

- Também é possível selecionar em uma lista de identificadores de controle disponíveis ao pressionar Ctrl+Barra de espaço. A lista contém os identificadores de controle seguidos dos itens disponíveis com a variável de conclusão automática do script.

Se inserir manualmente identificadores, retenha quaisquer espaços, já que todos os espaços nos identificadores são significativos.

No tempo de execução, e para todos os controles, exceto caixas de seleção, grupos de caixa de seleção e controle de texto estático, cada identificador é substituído com o valor atual da propriedade **Script** do controle associado. Se o controle estiver vazio no tempo de execução, ele não gerará qualquer script. Para caixas de seleção e grupos de caixas de seleção, o identificador é substituído pelo valor atual da propriedade Script R Verificado ou Script R Não Verificado do controle associado, dependendo do estado atual do controle - verificado ou não verificado. Veja o tópico “[Tipos de controle](#)” na página 36 para obter mais informações.

Exemplo: Incluindo valores de tempo de execução em um modelo de script R

Neste exemplo, o diálogo customizado do nó gerará e executar o script R para construir e escorar um modelo de regressão linear usando uma chamada para a função R `lm` com a assinatura mostrada aqui.

```
lm(formula,data)
```

- *formula* especifica uma expressão, como `Na~Age`, em que `Na` é o campo de destino do modelo e o campo de entrada do modelo é `Age`.
- *data* é o quadro de dados que contém os valores dos campos especificados na fórmula.

Considere um diálogo customizado do nó com um único controle do seletor de campo que permita que o usuário escolha o campo de entrada do modelo linear. O modelo de script para gerar e executar o script R que constrói o modelo é inserido na guia **Script**, e pode ficar parecido com este:

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

- `%>%input%%` é o valor da propriedade Identificador para o controle do seletor de campo. No tempo de execução, ele será substituído pelo valor atual da propriedade **Script** do controle.
- Definir a propriedade **Script** do controle de seletor de campo `%%ThisValue%%` especifica que no tempo de execução, o valor atual da propriedade será o valor do controle, que é o campo escolhido no seletor de campo.

Suponha que o usuário do diálogo customizado do nó selecione o campo `Idade` como o campo de entrada do modelo. O seguinte script R será então gerado pelo diálogo do nó:

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

O modelo do script para gerar e executar o script R que escora o modelo é inserido na guia **Script de Escoragem** e pode se assemelhar a:

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

Este script R não depende de nenhum valor especificado pelo usuário, somente do modelo construído usando o script R de construção do modelo. Portanto, o script R de escoragem de modelo é inserido como deveria ser no campo **Sintaxe de escoragem do modelo R** do nó Construção R.

Visualizando um diálogo de nó customizado

É possível visualizar o diálogo de nó que está atualmente aberto no Construtor de diálogo personalizado. O diálogo aparece e funciona como se estivesse executando a partir de um nó dentro do IBM SPSS Modeler.

- Os seletores de campo são preenchidos com campos simulados.
- O botão **OK** fecha a visualização.
- Se um arquivo de ajuda for especificado, o botão **Ajuda** será ativado e abrirá o arquivo especificado. Se nenhum arquivo de ajuda for especificado, o botão Ajuda será desativado quando houver visualização e oculto quando o diálogo real for executado.

Para visualizar um diálogo de nó customizado, a partir dos menus no Construtor de diálogo personalizado, escolha **Arquivo > Diálogo de visualização**.

Tipos de controle

A paleta de ferramentas fornece todos os controles padrão que podem ser necessários em um diálogo de nó customizado.

- **Seletor de campo:** uma lista de todos os campos do conjunto de dados ativo. Veja o tópico “[Seletor de campo](#)” na página 37 para obter mais informações.
- **Caixa de seleção:** uma única caixa de seleção. Veja o tópico “[Caixa de seleção](#)” na página 38 para obter mais informações.
- **Caixa de combinação:** uma caixa de combinação para criação de listas suspensas. Veja o tópico “[Caixa de combinação](#)” na página 38 para obter mais informações.
- **Caixa de listagem:** uma caixa de listagem para criação de listas de seleção única ou múltipla. Veja o tópico “[Caixa de combinação](#)” na página 38 para obter mais informações.
- **Controle de texto:** uma caixa de texto que aceita texto arbitrário como entrada. Veja o tópico “[Controle de texto](#)” na página 41 para obter mais informações.
- **Controle de número:** uma caixa de texto que é restrita aos valores numéricos como entrada. Veja o tópico “[Controle de número](#)” na página 42 para obter mais informações.
- **Controle de data:** um controle Spinner para especificar os valores de data/hora, que inclui datas, horas e datas/horas. Veja o tópico “[Controle de data](#)” na página 43 para obter mais informações.
- **Texto assegurado:** uma caixa de texto que mascara a entrada do usuário com asteriscos. Veja o tópico “[Texto protegido](#)” na página 44 para obter mais informações.
- **Controle Texto Estático:** um controle para exibição de texto estático. Veja o tópico “[Controle de texto estático](#)” na página 45 para obter mais informações.
- **Selecionador de cor:** um controle para especificação de uma cor e geração do valor RGB associado. Veja o tópico “[Selecionador de cor](#)” na página 46 para obter mais informações.
- **Controle de tabela:** uma tabela com um número fixo de colunas e um número variável de linhas que são incluídas no tempo de execução. Veja o tópico “[Controle de tabela](#)” na página 46 para obter mais informações.
- **Grupo de itens:** um contêiner para agrupamento de um conjunto de controles, como um conjunto de caixas de seleção. Veja o tópico “[Grupo de itens](#)” na página 48 para obter mais informações.
- **Grupo de opções:** um grupo de botões de opções. Veja o tópico “[Grupo de opções](#)” na página 49 para obter mais informações.
- **Grupo de caixas de seleção:** um contêiner para um conjunto de controles que são ativados ou desativados como um grupo, por uma caixa de seleção única. Veja o tópico “[Grupo de caixa de seleção](#)” na página 50 para obter mais informações.
- **Navegador do arquivo:** um controle para navegar pelo sistema de arquivos para abrir ou salvar um arquivo. Veja o tópico “[Navegador de arquivo](#)” na página 51 para obter mais informações.
- **Guia:** uma única guia. Veja o tópico “[Guia](#)” na página 52 para obter mais informações.

- **Botão de subdiálogo:** um botão para ativação de um subdiálogo. Veja o tópico “[Botão de subdiálogo](#)” na página 52 para obter mais informações.

Seletor de campo

O controle do Seletor de Campo exibe a lista de campos disponíveis para o usuário final do diálogo nó. É possível exibir todos os campos do conjunto de dados ativo (o padrão) ou é possível filtrar a lista com base no tipo e nível de medição, por exemplo, campos numéricos que possuem nível de medição de escala. Também é possível especificar qualquer outro controle de Seletor de Campo como os campos de origem para o Seletor de Campo. O controle de Seletor de Campo tem as propriedades a seguir:

Identificador. O identificador exclusivo para o controle.

Título. Um título opcional que aparece acima do controle. Para títulos com várias linhas, use ' \n para especificar quebras de linha.

Rankeamento do Título. Especifica a posição do título em relação ao controle. Valores são Parte Superior e Esquerda onde Parte Superior é o padrão. Essa propriedade aplica-se somente quando o tipo de seletor estiver configurado para selecionar um campo único.

ToolTip. O texto da Dica de ferramenta opcional que aparece quando o usuário passa o mouse sobre o controle. O texto especificado aparece apenas ao passar o mouse sobre a área de título do controle. Passar o mouse sobre um dos campos listados exibe o nome e o rótulo do campo.

Tecla Mnemônica. Um caractere opcional no título a ser usado como um atalho de teclado para o controle. O caractere aparece sublinhado no título. O atalho é ativado ao pressionar Alt+[tecla mnemônica].

Tipo de Seletor. Especifica se o Seletor de Campo no diálogo do nó customizado pode ser usado para selecionar um campo único ou vários campos a partir da lista de campos.

Tipo de Separador. Especifica o delimitador entre os campos selecionados no script gerado. Os separadores permitidos são um espaço em branco, uma vírgula e um sinal de mais (+). Também é possível inserir um caractere único arbitrário para ser usado como o separador.

Campos mínimos. O número mínimo de campos que devem ser especificados para o controle, se houver.

Campos máximos. O número máximo de campos que podem ser especificados para o controle, se houver.

Necessário para execução. Especifica se é necessário um valor nesse controle para que a execução prossiga. Se **True** for especificado, o usuário do diálogo do nó deverá especificar um valor para o controle, caso contrário, clicar em **OK** gerará um erro. Se **False** for especificado, a ausência de um valor nesse controle não terá nenhum efeito no estado do botão **OK**.

Filtro de variável. Permite que você filtre o conjunto de campos que são exibidos no controle. É possível filtrar em tipo de campo e nível de medição e é possível especificar que conjuntos de múltiplas respostas são incluídos na lista de campos. Clique no botão de reticências (...) para abrir o diálogo Filtro. Também é possível abrir o diálogo de Filtro clicando duas vezes no controle de Seletor de Campo nas telas. Veja o tópico “[Filtrando Listas de Campos](#)” na página 38 para obter mais informações.

Fonte de Campo. Especifica que um outro controle de Seletor de Campo é a origem de campos para o Seletor de Campo atual. Quando a propriedade de Fonte de Campo não estiver configurada, a fonte de campos é a conjunto de dados ativo. Clique no botão reticências (...) para abrir a caixa de diálogo e especifique a fonte de campo.

Script. Especifica o script da que é gerado e executado por esse controle no tempo de execução e pode ser inserido no modelo script.

- É possível especificar qualquer script R ou script Python para Spark válido. Para scripts de multilinhas ou scripts de longos, clique no botão reticências (...) e insira seu script de no diálogo da Propriedade do Script de .
- O valor `%%ThisValue%%` especifica o valor do tempo de execução do controle, que é a lista de campos. Este é o padrão.

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Especificando a Fonte de Campo para um Seletor de Campo

A caixa de diálogo Fonte de Campo especifica a origem dos campos que são exibidos no Seletor de Campo . A origem pode ser qualquer outro Seletor de Campo. É possível optar por exibir os campos que estiverem no controle selecionado ou os campos, a partir do conjunto de dados ativo, que não estiverem no controle selecionado.

Filtrando Listas de Campos

A caixa de diálogo Filtro, associada aos controles de seletor de campo, permite filtrar os tipos de campos a partir do conjunto de dados ativo que possam aparecer nas listas. Também é possível especificar se conjuntos de múltiplas respostas associados ao conjunto de dados ativo são incluídos. Campos numéricos incluem todos os formatos numéricos, exceto os formatos de data e hora.

Caixa de seleção

O controle Caixa de seleção é uma caixa de seleção simples que pode gerar e executar diferentes scripts R ou scripts Python for Spark para o estado marcado versus desmarcado. O controle Caixa de seleção possui as propriedades a seguir:

Identificador. O identificador exclusivo para o controle.

Título. Um título opcional que aparece acima do controle. Para títulos com várias linhas, use ' \n para especificar quebras de linha.

ToolTip. O texto da Dica de ferramenta opcional que aparece quando o usuário passa o mouse sobre o controle.

Tecla Mnemônica. Um caractere opcional no título a ser usado como um atalho de teclado para o controle. O caractere aparece sublinhado no título. O atalho é ativado ao pressionar Alt+[tecla mnemônica].

Valor Padrão. O estado padrão da caixa de seleção, marcado ou desmarcado.

Sintaxe marcada/desmarcadaScript. Especifica a script R ou script Python for Spark que é gerada e executada quando o controle é marcado e quando ele é desmarcado. Para incluir o script no modelo script, use o valor da propriedade Identifier. O gerado script, se a partir da propriedade Checked Script ou Uncheck Script , será inserida nas posições especificadas do identificador. Por exemplo, se o identificador for checkbox1, então, no tempo de execução, as instâncias de %%checkbox1%% no modelo script serão substituídas pelo valor da propriedade Script Marcada quando a caixa está marcada e a propriedade Script Desmarcada quando a caixa está desmarcada.

- É possível especificar qualquer script R ou script Python para Spark válido. Para scripts de multilinhas ou scripts de longos, clique no botão reticências (...) e insira seu script de no diálogo da Propriedade do Script de .

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Caixa de combinação

O controle Caixa de Combinação permite criar uma lista suspensa que pode gerar e executar script R ou script Python for Spark específico para o item de lista selecionado. É limitado a seleção única. O controle Caixa de Combinação possui as propriedades a seguir:

Identificador. O identificador exclusivo para o controle. Este é o identificador a ser usado ao referenciar o controle no modelo script.

Título. Um título opcional que aparece acima do controle. Para títulos com várias linhas, use ' \n para especificar quebras de linha.

Posição do título. Especifica a posição do título em relação ao controle. Valores são Parte Superior e Esquerda onde Parte Superior é o padrão.

ToolTip. O texto da Dica de ferramenta opcional que aparece quando o usuário passa o mouse sobre o controle.

Itens da lista. Clique no botão de reticências (...) para abrir a caixa de diálogo Propriedades de itens da lista, que permite especificar os itens de lista do controle. Também é possível abrir o diálogo Propriedades do Item da Lista dando um clique duplo no controle Caixa de Combinação na tela.

Tecla Mnemônica. Um caractere opcional no título a ser usado como um atalho de teclado para o controle. O caractere aparece sublinhado no título. O atalho é ativado ao pressionar Alt+[tecla mnemônica].

Editável. Especifica se o controle Caixa de Combinação é editável. Quando o controle é editável, um valor customizado pode ser inserido no tempo de execução.

Script. Especifica o script R ou Python for Spark que é gerado por esse controle no tempo de execução e pode ser inserido no modelo de script.

- O valor %%ThisValue%% especifica o valor do tempo de execução do controle e é o padrão. Se os itens da lista forem definidos manualmente, o valor do tempo de execução será o valor da propriedade Script para o item da lista selecionado. Se os itens da lista forem baseados em um controle de lista de metas, o valor do tempo de execução será o valor do item da lista selecionado. Para controles de caixa de listagem de seleção múltipla, o valor do tempo de execução será uma lista separada por espaços em branco dos itens selecionados. Consulte o tópico [“Especificando itens de lista para caixas de combinação e caixas de listagem”](#) na página 40 para obter mais informações.
- É possível especificar qualquer script R ou script Python para Spark válido. Para scripts de multilinhas ou scripts de longos, clique no botão reticências (...) e insira seu script de no diálogo da Propriedade do Script de .

Manipulação de aspas. Especifica o tratamento de aspas no valor de tempo de execução de ' %%%ThisValue%%%' quando a propriedade ' 'Script contém ' %%ThisValue%%%' como parte de uma cadeia de caracteres entre aspas. Neste contexto, uma sequência de caracteres entre aspas é uma sequência que está entre aspas simples ou aspas duplas. O tratamento das aspas aplica-se apenas às aspas que são do mesmo tipo que as aspas que cercam %%ThisValue%%%. Os tipos de manipulação de aspas a seguir estão disponíveis:

Python

As aspas no valor de tempo de execução de ' %%ThisValue%%%' que coincidem com as aspas adjacentes são escapadas com o caractere de barra invertida (\). Por exemplo, se a propriedade " "Script for " '%%ThisValue%' ' e o valor de tempo de execução da caixa de combinação for " Combo_box's value, o " "script gerado será " 'Combo_box\' s value'. Observe que o tratamento não é feito quando %%ThisValue%% é delimitado por aspas triplas.

R

As aspas no valor de tempo de execução de ' %%ThisValue%%%' que coincidem com as aspas adjacentes são escapadas com o caractere de barra invertida (\). Por exemplo, se a propriedade " "Script for " '%%ThisValue%' ' e o valor de tempo de execução da caixa de combinação for " Combo_box's value, o " "script gerado será " 'Combo_box\' s value'.

Nenhum

As aspas no valor de tempo de execução de %%ThisValue%% que correspondem a aspas delimitadoras são retidas sem modificação.

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica

visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Especificando itens de lista para caixas de combinação e caixas de listagem

A caixa de diálogo Propriedades do Item da Lista permite especificar os itens de lista de um controle de caixa de combinação ou de caixa de listagem.

Valores definidos manualmente. Permite especificar explicitamente cada um dos itens de lista.

- **Identificador.** Um identificador exclusivo para o item da lista.
- **NOME.** O nome que aparece na lista para esse item. O nome é um campo obrigatório.
- **Padrão.** Para uma caixa de combinação, especifica se o item da lista é o item padrão exibido na caixa de combinação. Para uma caixa de listagem, especifica se o item da lista é selecionado por padrão.
- **Script.** Especifica o script R ou script Python para Spark que é gerado quando o item da lista é selecionado.
- É possível especificar qualquer script R ou script Python para Spark válido. Para scripts de multilinhas ou scripts de longos, clique no botão reticências (...) e insira seu script de no diálogo da Propriedade do Script de .

Nota: É possível incluir um novo item da lista na linha em branco na parte inferior da lista existente.

Inserir qualquer uma das propriedades diferentes do identificador gera um identificador exclusivo, que poderá ser mantido ou modificado. É possível excluir um item da lista ao clicar na célula *Identificador* do item e pressionar Excluir.

Caixa de listagem

O controle da Lista de Lista permite exibir uma lista de itens que suportam seleção única ou múltipla e gerar R script ou Python para script Spark específicos para os itens selecionados. O controle Caixa de Listagem possui as propriedades a seguir:

Identificador. O identificador exclusivo para o controle. Este é o identificador a ser usado ao referenciar o controle no modelo script.

Título. Um título opcional que aparece acima do controle. Para títulos com várias linhas, use '\n' para especificar quebras de linha.

ToolTip. O texto da Dica de ferramenta opcional que aparece quando o usuário passa o mouse sobre o controle.

Itens da lista. Clique no botão de reticências (...) para abrir a caixa de diálogo Propriedades de itens da lista, que permite especificar os itens de lista do controle. Também é possível abrir o diálogo Propriedades do Item da Lista dando um clique duplo no controle Caixa de Listagem na tela.

Tecla Mnemônica. Um caractere opcional no título a ser usado como um atalho de teclado para o controle. O caractere aparece sublinhado no título. O atalho é ativado ao pressionar Alt+[tecla mnemônica].

Tipo de caixa de listagem. Especifica se a caixa de listagem suporta apenas seleção única ou seleção múltipla. Também é possível especificar que os itens sejam exibidos como uma lista de caixas de seleção.

Tipo de Separador. Especifica o delimitador entre os itens de lista selecionados na script gerada. Os separadores permitidos são um espaço em branco, uma vírgula e um sinal de mais (+). Também é possível inserir um caractere único arbitrário para ser usado como o separador.

Mínimo selecionado. O número mínimo de itens que devem ser selecionados no controle, se houver.

Máximo selecionado. O número máximo de itens que podem ser selecionados no controle, se houver.

Script. Especifica o script R ou Python for Spark que é gerado por esse controle no tempo de execução e pode ser inserido no modelo de script.

- O valor %%ThisValue%% especifica o valor do tempo de execução do controle e é o padrão. Se os itens da lista forem definidos manualmente, o valor do tempo de execução será o valor da propriedade

Script para o item da lista selecionado. Se os itens da lista forem baseados em um controle de lista de metas, o valor do tempo de execução será o valor do item da lista selecionado. Para vários controles de lista de seleção, o valor do tempo de execução é uma lista dos itens selecionados, delimitada pelo Tipo de Separador especificado (o padrão é separado por espaços em branco). Consulte o tópico “[Especificando itens de lista para caixas de combinação e caixas de listagem](#)” na página 40 para obter mais informações.

- É possível especificar qualquer script R ou script Python para Spark válido. Para scripts de multilinhas ou scripts de longos, clique no botão reticências (...) e insira seu script de no diálogo da Propriedade do Script de .

Manipulação de aspas. Especifica o tratamento de aspas no valor de tempo de execução de '%ThisValue%%' quando a propriedade 'Script' contém '%ThisValue%%' como parte de uma cadeia de caracteres entre aspas. Neste contexto, uma sequência de caracteres entre aspas é uma sequência que está entre aspas simples ou aspas duplas. O tratamento das aspas aplica-se apenas às aspas que são do mesmo tipo que as aspas que cercam '%ThisValue%%'. Os tipos de manipulação de aspas a seguir estão disponíveis:

Python

As aspas no valor de tempo de execução de '%ThisValue%%' que correspondem às aspas adjacentes são escapadas com o caractere de barra invertida (\). Por exemplo, se a propriedade 'Script' for '%ThisValue%' e o item de lista selecionado for 'List item's value', então o script gerado será 'List item\ 's value'. Observe que o tratamento não é feito quando '%ThisValue%%' é delimitado por aspas triplas.

R

As aspas no valor de tempo de execução de '%ThisValue%%' que correspondem às aspas adjacentes são escapadas com o caractere de barra invertida (\). Por exemplo, se a propriedade 'Script' for '%ThisValue%' e o item de lista selecionado for 'List item's value', então o script gerado será 'List item\ 's value'.

Nenhum

As aspas no valor de tempo de execução de '%ThisValue%%' que correspondem a aspas delimitadoras são retidas sem modificação.

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Controle de texto

O controle Texto é uma caixa de texto simples que pode aceitar entrada arbitrária e possui as propriedades a seguir:

Identificador. O identificador exclusivo para o controle. Este é o identificador a ser usado ao referenciar o controle no modelo script.

Título. Um título opcional que aparece acima do controle. Para títulos com várias linhas, use '\n' para especificar quebras de linha.

Posição do título. Especifica a posição do título em relação ao controle. Valores são Parte Superior e Esquerda onde Parte Superior é o padrão.

ToolTip. O texto da Dica de ferramenta opcional que aparece quando o usuário passa o mouse sobre o controle.

Tecla Mnemônica. Um caractere opcional no título a ser usado como um atalho de teclado para o controle. O caractere aparece sublinhado no título. O atalho é ativado ao pressionar Alt+[tecla mnemônica].

Conteúdo de texto. Especifica se os conteúdos são arbitrários ou se a caixa de texto deve conter uma sequência de caracteres que adira às regras para nomes de campos do IBM SPSS Modeler.

Valor Padrão. Os conteúdos padrão da caixa de texto.

Largura. Especifica a largura da área de texto do controle em caracteres. Os valores permitidos são números inteiros positivos. Qualquer valor vazio significa que a largura será automaticamente determinada.

Requerido para execução. Especifica se é necessário um valor nesse controle para que a execução prossiga. Se **True** for especificado, o usuário do diálogo do nó deverá especificar um valor para o controle, caso contrário, clicar em **OK** gerará um erro. Se **False** for especificado, a ausência de um valor nesse controle não terá nenhum efeito no estado do botão **OK**. O padrão é **False**.

Script. Especifica a script R ou script Python for Spark que é gerado e executado por este controle no tempo de execução e pode ser inserido no modelo script.

- É possível especificar qualquer script R ou script Python para Spark válido. Para scripts de multilinhas ou scripts de longos, clique no botão reticências (...) e insira seu script de no diálogo da Propriedade do Script de .
- O valor `%%ThisValue%%` especifica o valor de tempo de execução do controle, que é o conteúdo da caixa de texto. Este é o padrão.
- Se a propriedade " "Script incluir " `%%ThisValue%%` e o valor de tempo de execução da caixa de texto estiver vazio, o controle da caixa de texto não gerará nenhum "script".

Manipulação de aspas. Especifica o tratamento de aspas no valor de tempo de execução de '`%%ThisValue%%`' quando a propriedade ' 'Script contém '`%%ThisValue%%`' como parte de uma cadeia de caracteres entre aspas. Neste contexto, uma sequência de caracteres entre aspas é uma sequência que está entre aspas simples ou aspas duplas. O tratamento das aspas aplica-se apenas às aspas que são do mesmo tipo que as aspas que cercam `%%ThisValue%%`. Os tipos de manipulação de aspas a seguir estão disponíveis:

Python

As aspas no valor de tempo de execução de '`%%ThisValue%%`' que coincidem com as aspas adjacentes são escapadas com o caractere de barra invertida (\). Por exemplo, se a propriedade ' 'Script for' '`%%ThisValue%%`' e o valor de tempo de execução do controle de texto for 'Text box's value, então o 'script gerado será ' 'Text box\'s value'. O tratamento de aspas não é feito quando `%%ThisValue%%` é delimitado por aspas triplas.

R

As aspas no valor de tempo de execução de '`%%ThisValue%%`' que coincidem com as aspas adjacentes são escapadas com o caractere de barra invertida (\). Por exemplo, se a propriedade ' 'Script for' '`%%ThisValue%%`' e o valor de tempo de execução do controle de texto for 'Text box's value, então o 'script gerado será ' 'Text box\'s value'.

Nenhum

Aspas no valor de tempo de execução de '`%%ThisValue%%`' que correspondem a aspas delimitadoras são retidas sem modificação.

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Controle de número

O controle Número é uma caixa de texto para inserção de um valor numérico e possui as propriedades a seguir:

Identificador. O identificador exclusivo para o controle. Este é o identificador a ser usado ao referenciar o controle no modelo script.

Título. Um título opcional que aparece acima do controle. Para títulos com várias linhas, use '\n' para especificar quebras de linha.

Posição do título. Especifica a posição do título em relação ao controle. Valores são Parte Superior e Esquerda onde Parte Superior é o padrão.

ToolTip. O texto da Dica de ferramenta opcional que aparece quando o usuário passa o mouse sobre o controle.

Tecla Mnemônica. Um caractere opcional no título a ser usado como um atalho de teclado para o controle. O caractere aparece sublinhado no título. O atalho é ativado ao pressionar Alt+[tecla mnemônica].

Tipo numérico. Especifica qualquer limitação sobre o que pode ser inserido. Um valor de Real especifica que não há restrições no valor inserido, diferente do que seria com um valor numérico. Um valor de Número inteiro especifica que o valor deve ser um número inteiro.

Entrada Spin. Especifica se o controle é exibido como um spinner. O padrão é False.

Incremento. O incremento quando o controle é exibido como um spinner.

Valor Padrão. O valor padrão, se houver.

Valor mínimo. O valor mínimo permitido, se houver.

Valor máximo. O valor máximo permitido, se houver.

Largura. Especifica a largura da área de texto do controle em caracteres. Os valores permitidos são números inteiros positivos. Qualquer valor vazio significa que a largura será automaticamente determinada.

Requerido para execução. Especifica se é necessário um valor nesse controle para que a execução prossiga. Se **True** for especificado, o usuário do diálogo do nó deverá especificar um valor para o controle, caso contrário, clicar em **OK** gerará um erro. Se **False** for especificado, a ausência de um valor nesse controle não terá nenhum efeito no estado do botão **OK**. O padrão é **False**.

Script. Especifica a script R ou script Python for Spark que é gerado e executado por este controle no tempo de execução e pode ser inserido no modelo script.

- É possível especificar qualquer script R ou script Python para Spark válido. Para scripts de multilinhas ou scripts de longos, clique no botão reticências (...) e insira seu script de no diálogo da Propriedade do Script de .
- O valor `%%ThisValue%%` especifica o valor do tempo de execução do controle, que é o valor numérico. Este é o padrão.
- Se a propriedade " "Script incluir " `%%ThisValue%%` e o valor de tempo de execução do controle numérico estiver vazio, o controle numérico não gerará nenhum " "script.

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Controle de data

O controle Data é um controle Spinner para especificar valores de data/hora, que incluem datas, horas e datas/horas. O controle Data possui as propriedades a seguir:

Identificador. O identificador exclusivo para o controle. Este é o identificador a ser usado ao referenciar o controle no modelo script.

Título. Um título opcional que aparece acima do controle. Para títulos com várias linhas, use ' \n para especificar quebras de linha.

Posição do título. Especifica a posição do título em relação ao controle. Valores são Parte Superior e Esquerda onde Parte Superior é o padrão.

ToolTip. O texto da Dica de ferramenta opcional que aparece quando o usuário passa o mouse sobre o controle.

Tecla Mnemônica. Um caractere opcional no título a ser usado como um atalho de teclado para o controle. O caractere aparece sublinhado no título. O atalho é ativado ao pressionar Alt+[tecla mnemônica].

Tipo. Especifica se o controle é para data, horas ou valores de data/hora.

Data

O controle especifica uma data do calendário no formato aaaa-mm-dd. O valor de tempo de execução padrão é especificado pela propriedade Valor Padrão.

Hora

O controle especifica o horário do dia no formato hh:mm:ss. O valor de tempo de execução padrão é o horário do dia atual.

Data/hora

O controle especifica uma data e hora no formato aaaa-mm-dd hh:mm:ss. O valor de tempo de execução padrão é a data e hora do dia atual.

Valor Padrão. O valor do tempo de execução padrão do controle quando o tipo é Data. É possível especificar a exibição da data atual ou de uma data em particular.

Script. Especifica a script R ou script Python for Spark que é gerado e executado por este controle no tempo de execução e pode ser inserido no modelo script.

- É possível especificar qualquer script R ou script Python para Spark válido. Para scripts de multilinhas ou scripts de longos, clique no botão reticências (...) e insira seu script de no diálogo da Propriedade do Script de .
- O valor %%ThisValue%% especifica o valor de tempo de execução do controle. Este é o padrão.

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Nota: O controle Data não é suportado em liberações do IBM SPSS Modeler anteriores à liberação 18.

Texto protegido

O controle Texto Assegurado é uma caixa de texto que mascara a entrada do usuário com asteriscos.

Identificador. O identificador exclusivo para o controle. Este é o identificador a ser usado ao referenciar o controle no modelo script.

Título. Um título opcional que aparece acima do controle. Para títulos com várias linhas, use ' \n para especificar quebras de linha.

Posição do título. Especifica a posição do título em relação ao controle. Valores são Parte Superior e Esquerda onde Parte Superior é o padrão.

ToolTip. O texto da Dica de ferramenta opcional que aparece quando o usuário passa o mouse sobre o controle.

Tecla Mnemônica. Um caractere opcional no título a ser usado como um atalho de teclado para o controle. O caractere aparece sublinhado no título. O atalho é ativado ao pressionar Alt+[tecla mnemônica].

Largura. Especifica a largura da área de texto do controle em caracteres. Os valores permitidos são números inteiros positivos. Qualquer valor vazio significa que a largura será automaticamente determinada.

Requerido para execução. Especifica se é necessário um valor nesse controle para que a execução prossiga. Se **True** for especificado, o usuário do diálogo do nó deverá especificar um valor para o controle, caso contrário, clicar em **OK** gerará um erro. Se **False** for especificado, a ausência de um valor nesse controle não terá nenhum efeito no estado do botão **OK**. O padrão é **False**.

Script. Especifica a script R ou script Python for Spark que é gerado e executado por este controle no tempo de execução e pode ser inserido no modelo script.

- É possível especificar qualquer script R ou script Python para Spark válido. Para scripts de multilinhas ou scripts de longos, clique no botão reticências (...) e insira seu script de no diálogo da Propriedade do Script de .
- O valor %%ThisValue%% especifica o valor de tempo de execução do controle, que é o conteúdo da caixa de texto. Este é o padrão.
- Se a propriedade " "Script incluir" %%ThisValue%% e o valor de tempo de execução do controle de texto protegido estiver vazio, o controle de texto protegido não gerará nenhum " "Script R ou ' Python para script Spark.

Manipulação de aspas. Especifica o tratamento de aspas no valor de tempo de execução de ' % %%ThisValue%% quando a propriedade ' 'Script contém ' %%ThisValue%% como parte de uma cadeia de caracteres entre aspas. Neste contexto, uma sequência de caracteres entre aspas é uma sequência que está entre aspas simples ou aspas duplas. O tratamento das aspas aplica-se apenas às aspas que são do mesmo tipo que as aspas que delimitam %%ThisValue%% e apenas quando Encrypt passed value=False. Os tipos de manipulação de aspas a seguir estão disponíveis:

Python

As aspas no valor de tempo de execução de ' %%ThisValue%% que coincidem com as aspas que as envolvem são escapadas com o caractere de barra invertida (\). Por exemplo, se a propriedade ' 'Script for' ' %%ThisValue%' e o valor de tempo de execução do controle for ' Secured Text's value, então o ' 'script gerado será ' 'Secured Text\'s value'. O tratamento de aspas não é feito quando %%ThisValue%% é delimitado por aspas triplas.

R

As aspas no valor de tempo de execução de ' %%ThisValue%% que coincidem com as aspas que as envolvem são escapadas com o caractere de barra invertida (\). Por exemplo, se a propriedade ' 'Script for' ' %%ThisValue%' e o valor de tempo de execução do controle for ' Secured Text's value, então o ' 'script gerado será ' 'Secured Text\'s value'.

Nenhum

Aspas no valor do tempo de execução de %%ThisValue%% que correspondem a aspas delimitadoras são retidas sem modificação.

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Nota: O controle de texto assegurado não é suportado em liberações do IBM SPSS Modeler anteriores à liberação 18.

Controle de texto estático

O controle de texto estático permite incluir um bloco de texto em seu diálogo nó e possui as propriedades a seguir:

Identificador. O identificador exclusivo para o controle.

TÍTULO. O conteúdo do bloco de texto. Para conteúdo multilinhas, use \n para especificar quebras de linha.

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Selecionador de cor

O controle Selecionador de Cor é uma interface com o usuário para especificar uma cor e gerar o valor RGB associado. O controle Selecionador de Cor possui as propriedades a seguir:

Identificador. O identificador exclusivo para o controle. Este é o identificador a ser usado ao referenciar o controle no modelo script.

Título. Um título opcional que aparece acima do controle. Para títulos com várias linhas, use ' \n para especificar quebras de linha.

Posição do título. Especifica a posição do título em relação ao controle. Valores são Parte Superior e Esquerda onde Parte Superior é o padrão.

ToolTip. O texto da Dica de ferramenta opcional que aparece quando o usuário passa o mouse sobre o controle.

Tecla Mnemônica. Um caractere opcional no título a ser usado como um atalho de teclado para o controle. O caractere aparece sublinhado no título. O atalho é ativado ao pressionar Alt+[tecla mnemônica].

Script. Especifica a script R ou script Python for Spark que é gerado e executado por este controle no tempo de execução e pode ser inserido no modelo script.

- É possível especificar qualquer script R ou script Python para Spark válido. Para scripts de multilinhas ou scripts de longos, clique no botão reticências (...) e insira seu script de no diálogo da Propriedade do Script de .
- O valor %%ThisValue%% especifica o valor do tempo de execução do controle, que é o valor RGB da cor selecionada. O valor RGB é representado como uma lista separada por espaços em branco de números inteiros na seguinte ordem: Valor R, valor G, valor B.

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Nota: O controle de selecionador de cor não é suportado em liberações do IBM SPSS Modeler anteriores à liberação 18.

Controle de tabela

O controle Tabela cria uma tabela com um número de linhas fixo e um número variável de linhas que são incluídas no tempo de execução. O controle Tabela possui as propriedades a seguir:

Identificador. O identificador exclusivo para o controle. Este é o identificador a ser usado ao referenciar o controle no modelo script.

Título. Um título opcional que aparece acima do controle. Para títulos com várias linhas, use ' \n para especificar quebras de linha.

ToolTip. O texto da Dica de ferramenta opcional que aparece quando o usuário passa o mouse sobre o controle.

Tecla Mnemônica. Um caractere opcional no título a ser usado como um atalho de teclado para o controle. O caractere aparece sublinhado no título. O atalho é ativado ao pressionar Alt+[tecla mnemônica].

Reordenar botões. Especifica se os botões mover para cima e mover para baixo são incluídos na tabela. Esses botões são usados no tempo de execução para reordenar as linhas da tabela.

Colunas da tabela. Clique no botão de reticências (...) para abrir a caixa de diálogo Colunas da tabela, na qual são especificadas as colunas da tabela.

Mínimo de linhas. O número mínimo de linhas que devem estar na tabela.

Máximo de linhas. O número máximo de linhas que podem estar na tabela.

Necessário para execução. Especifica se é necessário um valor nesse controle para que a execução prossiga. Se **True** for especificado, o usuário do diálogo do nó deverá especificar um valor para o controle, caso contrário, clicar em **OK** gerará um erro. Se **False** for especificado, a ausência de um valor nesse controle não terá nenhum efeito no estado do botão **OK**.

Script. Especifica o script R ou Python for Spark que é gerado por esse controle no tempo de execução e pode ser inserido no modelo de script.

- O valor `%%ThisValue%%` especifica o valor do tempo de execução do controle e é o padrão. O valor do tempo de execução é uma lista separada por espaços em branco da script que é gerada por cada coluna na tabela, iniciando com a coluna da extrema esquerda. Se a propriedade Script inclui `%%ThisValue%%` e nenhuma das colunas gerar script, então, a tabela como um todo não gera qualquer script.
- É possível especificar qualquer script R ou script Python para Spark válido. Para scripts de multilinhas ou scripts de longos, clique no botão reticências (...) e insira seu script de no diálogo da Propriedade do Script de .

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Nota: O controle Tabela não é suportado em liberações do IBM SPSS Modeler anteriores à liberação 18.

Especificando colunas para controles de tabela

A caixa de diálogo Colunas da Tabela especifica as propriedades das colunas do controle de Tabela.

Identificador. Um identificador exclusivo para a coluna.

Nome da Coluna. O nome da coluna conforme ele aparece na tabela.

Conteúdo. Especifica o tipo de dados para a coluna. O valor **Real** especifica que não há nenhuma restrição no valor inserido, além do fato de ele ser numérico. O valor **Número inteiro** especifica que o valor deve ser um número inteiro. O valor **Qualquer** especifica que não há nenhuma restrição no valor inserido. O valor **Nome de Variável** especifica que o valor deve atender aos requisitos para um nome de variável válido no IBM SPSS Estatísticas.

Valor Padrão. O valor padrão para essa coluna, se houver, quando novas linhas forem incluídas na tabela no tempo de execução.

Tipo de separador. Especifica o delimitador entre os valores da coluna no script gerado. Os separadores permitidos são um espaço em branco, uma vírgula e um sinal de mais (+). Também é possível inserir um caractere único arbitrário para ser usado como o separador.

Entre aspas. Especifica se cada valor na coluna é colocado entre aspas duplas no script gerado.

Manipulação de aspas. Especifica manipular aspas nas entradas da célula para a coluna quando a propriedade Entre Aspas for true. A manipulação de aspas se aplica somente nas aspas duplas nos valores da célula. Os tipos de manipulação de aspas a seguir estão disponíveis:

Python

As aspas duplas nos valores das células são escapadas com o caractere de barra invertida (\). Por exemplo, se o valor da célula for ' This "quoted" value, o 'script gerado será ' "This \ "quoted\ " value ".

R

As aspas duplas nos valores das células são escapadas com o caractere de barra invertida (\). Por exemplo, se o valor da célula for ' This "quoted" value, o 'script gerado será ' "This \ "quoted\ " value ".

Nenhum

As aspas duplas nos valores da célula são retidas sem nenhuma modificação.

Width(chars). Especifica a largura da coluna nos caracteres. Os valores permitidos são números inteiros positivos.

Script. Especifica o script R ou script Python para Spark que é gerado por essa coluna no tempo de execução. O script para a tabela como um todo é uma lista separada por espaços em branco do script que é gerado por cada coluna na tabela, iniciando com a coluna mais à esquerda.

- É possível especificar qualquer script R ou script Python para Spark válido. Para scripts de multilinhas ou scripts de longos, clique no botão reticências (...) e insira seu script de no diálogo da Propriedade do Script de .
- O valor %%ThisValue%% especifica o valor de tempo de execução da coluna, que é uma lista de valores na coluna delimitados pelo separador especificado.
- Se a propriedade " "Script da coluna incluir" %%ThisValue%% e o valor de tempo de execução da coluna estiver vazio, a coluna não gerará nenhum " "script.

Nota: É possível incluir uma linha para uma nova coluna da Tabela na linha em branco na parte inferior da lista existente no diálogo Colunas da Tabela. Inserir qualquer uma das propriedades diferentes do identificador gera um identificador exclusivo, que poderá ser mantido ou modificado. É possível excluir uma coluna da Tabela ao clicar na célula identificadora da coluna Tabela e pressionar Excluir.

Ligação para controle

É possível vincular um controle de Tabela a um controle de Seletor de Campo. Quando um controle de Tabela é vinculado a um Seletor de Campo, há uma linha na tabela para cada campo no Seletor de Campo. As linhas são incluídas na tabela ao incluir campos ao Seletor de campo. As linhas são excluídas da tabela ao remover os campos do Seletor de campo. Um controle de Tabela vinculado pode ser usado, por exemplo, para especificar as propriedades de campos que são selecionados em um Seletor de Campo.

Para ativar a vinculação, a tabela deve ter uma coluna com o Nome de Variável para a propriedade Conteúdos e deve haver pelo menos um controle de Seletor de Campo na tela.

Para vincular um controle de Tabela a um Seletor de Campo, especifique o Seletor de Campo na lista de Controles disponíveis no grupo Vincular a Controle na caixa de diálogo Colunas da Tabela. Em seguida, selecione a coluna da tabela, referida como a **Coluna vinculada**, que define o link. Quando a tabela estiver renderizada, a coluna vinculada exibe os campos atuais no Seletor de Campo. É possível vincular somente Seletores de Campo com vários campos.

Grupo de itens

O controle Grupo de Itens é um contêiner para outros controles, permitindo agrupar e controlar a script gerada a partir de vários controles. Por exemplo, você tem um conjunto de caixas de seleção que especifica configurações opcionais para um subcomando, mas quer apenas gerar a script para o subcomando se pelo menos uma caixa for selecionada. Isso é realizado usando um controle Grupo de Itens como um contêiner para os controles de caixa de seleção. Os tipos de controles a seguir podem ser contidos em um Grupo de itens: seletor de campo, caixa de seleção, caixa de combinação, caixa de listagem, controle de texto, controle de número, texto estático, grupo de opções e navegador de arquivos. O controle Grupo de Itens possui as propriedades a seguir:

Identificador. O identificador exclusivo para o controle. Este é o identificador a ser usado ao referenciar o controle no modelo script.

TÍTULO. Um título opcional para o grupo. Para títulos multilinhas, use \n para especificar quebras de linha.

Script. Especifica a script R ou script Python for Spark que é gerado e executado por este controle no tempo de execução e pode ser inserido no modelo script.

- É possível especificar qualquer script R ou script Python para Spark válido. Para scripts de multilinhas ou scripts de longos, clique no botão reticências (...) e insira seu script de no diálogo da Propriedade do Script de .
- É possível incluir identificadores para quaisquer controles contidos no grupo de itens. No tempo de execução, os identificadores são substituídos pela script R ou script Python gerada pelos controles.

- O valor %%ThisValue%% gera uma lista separada por espaços em branco da script R ou script Python gerada por cada controle no grupo de itens, na ordem em que aparecem no grupo (cima para baixo). Este é o padrão. Se a propriedade " "Script incluir" %%ThisValue%% e nenhum " "script for gerado por qualquer um dos controles no grupo de itens, o grupo de itens como um todo não gerará nenhum " "script.

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Grupo de opções

O controle Grupo de opções é um contêiner para um conjunto de botões de opções, cada um dos quais pode conter um conjunto de controlos aninhados.. O controle Grupo de Opções possui as propriedades a seguir:

Identificador. O identificador exclusivo para o controle. Este é o identificador a ser usado ao referenciar o controle no modelo script.

Título. Um título opcional para o grupo. Para títulos multilinhas, use \n para especificar quebras de linha.

ToolTip. O texto da Dica de ferramenta opcional que aparece quando o usuário passa o mouse sobre o controle.

Botões de opções. Clique no botão de reticências (...) para abrir a caixa de diálogo Propriedades do grupo de opções, que permite especificar as propriedades dos botões de opções, bem como incluir ou remover botões do grupo. A capacidade de aninhar controles em um determinado botão de opções é uma propriedade do botão de opções e é definido na caixa de diálogo Propriedades do grupo de opções. Observe que também é possível abrir o diálogo Propriedades do grupo de opções clicando duas vezes no controle Grupo de opções na tela.

Script. Especifica a script R ou script Python for Spark que é gerada por este controle no tempo de execução e pode ser inserido no modelo script.

- É possível especificar qualquer script R ou script Python para Spark válido. Para scripts de multilinhas ou scripts de longos, clique no botão reticências (...) e insira seu script de no diálogo da Propriedade do Script de .
- O valor %%ThisValue%% especifica o valor do tempo de execução do grupo de botão de opções, que é o valor da propriedade Script para o botão de opções selecionado. Este é o padrão. Se a propriedade " "Script incluir" %%ThisValue%% e nenhum " "script for gerado pelo botão de opção selecionado, o grupo de botões de opção não gerará nenhum " "script.

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Definindo os botões de opções

A caixa de diálogo Propriedades do Grupo de Botões de Opções permite especificar um grupo de botões de opções.

Identificador. Um identificador exclusivo para o botão de opções.

Nome da Coluna. O nome que aparece ao lado do botão de opções. O nome é um campo obrigatório.

ToolTip. O texto da Dica de ferramenta opcional que aparece quando o usuário passa o mouse sobre o controle.

Chave Mnemônica. Um caractere opcional no nome a ser usado como um mnemônico. O caractere especificado deve existir no nome.

Grupo Aninhado. Especifica se outros controles podem ser aninhados nesse botão de opções. O padrão é false. Quando a propriedade de grupo aninhado é configurada para true, uma área de lançamento retangular é exibida, aninhada e indentada no botão de opções associado. Os controles a seguir podem ser aninhados sob um botão de opções: seletor de campo, caixa de seleção, controle de texto, texto estático, controle de número, caixa de combinação, caixa de listagem e navegador de arquivos.

Padrão. Especifica se o botão de opções é a seleção padrão.

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Script. Especifica o script R ou script Python para Spark que é gerado quando o botão de opções é selecionado.

- É possível especificar qualquer script R ou script Python para Spark válido. Para scripts de multilinhas ou scripts de longos, clique no botão reticências (...) e insira seu script de no diálogo da Propriedade do Script de .
- Para botões de opções que contêm controles aninhados, o valor de %%ThisValue%% gera uma lista separada por espaço em branco do script R ou Python para Spark gerado por cada controle aninhado, na ordem em que eles aparecem no botão de opções (superior para inferior).

É possível incluir um novo botão de opções na linha em branco na parte inferior da lista existente. Inserir qualquer uma das propriedades diferentes do identificador gera um identificador exclusivo, que poderá ser mantido ou modificado. É possível excluir um botão de opções clicando na célula *Identificador* do botão e pressionando excluir.

Grupo de caixa de seleção

O controle Grupo de caixa de seleção é um contêiner para um conjunto de controles que são ativados ou desativados como um grupo, por uma caixa de seleção única. Os tipos de controle a seguir podem estar contidos em um Grupo de caixa de seleção: seletor de campo, caixa de seleção, caixa de combinação, caixa de listagem, controle de texto, controle de número, texto estático, grupo de opções e navegador de arquivos. O controle Grupo de caixa de seleção possui as propriedades a seguir:

Identificador. O identificador exclusivo para o controle. Este é o identificador a ser usado ao referenciar o controle no modelo script.

TÍTULO. Um título opcional para o grupo. Para títulos multilinhas, use \n para especificar quebras de linha.

Título da caixa de seleção. Um rótulo opcional que é exibido com a caixa de seleção de controle. Supora \n para especificar as quebras de linha.

ToolTip. O texto da Dica de ferramenta opcional que aparece quando o usuário passa o mouse sobre o controle.

Chave Mnemônica. Um caractere opcional no título a ser usado como um atalho de teclado para o controle. O caractere aparece sublinhado no título. O atalho é ativado ao pressionar Alt+[tecla mnemônica].

Valor Padrão. O estado padrão da caixa de seleção de controle, marcada ou desmarcada.

Script R Marcado/desmarcado. Especifica o script R que é gerado quando o controle é marcado e quando ele é desmarcado. Para incluir a script R no modelo script, use o valor da propriedade Identificador. A gerada R script, se a partir da propriedade Checked R Script ou Uncheck R Script , será inserida nas posições especificadas do identificador. Por exemplo, se o identificador for checkboxgroup1, então, no tempo de execução, as instâncias de %%checkboxgroup1%% no modelo script serão substituídas pelo valor da propriedade Script R marcada quando a caixa está marcada e a propriedade Script R desmarcada quando a caixa está desmarcada.

- É possível especificar qualquer script R ou script Python para Spark válido. Para scripts de multilinhas ou scripts de longos, clique no botão reticências (...) e insira seu script de no diálogo da Propriedade do Script de .
- É possível incluir identificadores para quaisquer controles contidos no grupo de caixa de seleção. No tempo de execução, os identificadores são substituídos pela script R gerada pelos controles.
- O valor `%%ThisValue%%` pode ser usado na propriedade Checked R Script ou Uncheck R Script . Ele gera uma lista separada por espaços em branco da script R gerada por cada controle no grupo de caixa de seleção, na ordem em que aparecem no grupo (cima para baixo).
- Por padrão, a propriedade Checked R Script tem um valor de `%%ThisValue%%` e a propriedade Uncheck R Script está em branco.

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Navegador de arquivo

O controle Navegador de Arquivo consiste de uma caixa de texto para um caminho de arquivo e um botão Navegar que abre um diálogo padrão do IBM SPSS Modeler para abrir ou salvar um arquivo. O controle Navegador de Arquivo possui as propriedades a seguir:

Identificador. O identificador exclusivo para o controle. Este é o identificador a ser usado ao referenciar o controle no modelo script.

Título. Um título opcional que aparece acima do controle. Para títulos com várias linhas, use ' \n para especificar quebras de linha.

Posição do título. Especifica a posição do título em relação ao controle. Valores são Parte Superior e Esquerda onde Parte Superior é o padrão.

ToolTip. O texto da Dica de ferramenta opcional que aparece quando o usuário passa o mouse sobre o controle.

Tecla Mnemônica. Um caractere opcional no título a ser usado como um atalho de teclado para o controle. O caractere aparece sublinhado no título. O atalho é ativado ao pressionar Alt+[tecla mnemônica].

Operação do sistema de arquivos. Especifica se o diálogo ativado pelo botão Navegar é apropriado para a abertura de arquivos ou para salvamento de arquivos. Um valor de Abrir indica que o diálogo Navegar valida a existência do arquivo especificado. Um valor de Salvar indica que o diálogo Navegar não valida a existência do arquivo especificado.

Tipo de navegador. Especifica se o diálogo Navegar é usado para selecionar um arquivo (Localizar arquivo) ou selecionar uma pasta (Localizar pasta).

Filtro de arquivo. Clique no botão de reticências (...) para abrir a caixa de diálogo Filtro de arquivo, que permite especificar os tipos de arquivo disponíveis para a caixa de diálogo Abrir ou Salvar. Por padrão, todos os tipos de arquivo são permitidos. Observe que também é possível abrir o diálogo Filtro de Arquivo clicando duas vezes no controle Navegador de Arquivo na tela.

Tipo de sistema de arquivos. No modo de análise distribuído, isto especifica se o diálogo Abrir ou Salvar procura o sistema de arquivos no qual o IBM SPSS Modeler Server está executando ou o sistema de arquivos de seu computador local. Selecione **Servidor** para procurar o sistema de arquivos do servidor ou **Cliente** para procurar o sistema de arquivos do seu computador local. A propriedade não tem efeito no modo de análise local.

Requerido para execução. Especifica se é necessário um valor nesse controle para que a execução prossiga. Se **True** for especificado, o usuário do diálogo do nó deverá especificar um valor para o controle, caso contrário, clicar em **OK** gerará um erro. Se **False** for especificado, a ausência de um valor nesse controle não terá nenhum efeito no estado do botão **OK**. O padrão é **False**.

Padrão. O valor padrão do controle.

Script. Especifica a script R ou script Python for Spark que é gerada por este controle no tempo de execução e pode ser inserido no modelo script.

- É possível especificar qualquer script R ou script Python para Spark válido. Para scripts de multilinhas ou scripts de longos, clique no botão reticências (...) e insira seu script de no diálogo da Propriedade do Script de .
- O valor %%ThisValue%% especifica o valor do tempo de execução da caixa de texto, que é o caminho de arquivo delimitado por aspas duplas, especificado manualmente ou preenchido pelo diálogo Navegar. Este é o padrão.
- Se a propriedade ' 'Script incluir ' %%ThisValue%% e o valor de tempo de execução da caixa de texto estiver vazio, o controle do navegador de arquivos não gerará nenhum 'script.

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Filtro de tipo de arquivo

A caixa de diálogo Filtro de Arquivo permite especificar os tipos de arquivo exibidos nas listas suspensas Arquivos de tipo e Salvar como tipo para abrir e salvar os diálogos acessados a partir de um controle de Navegador de Sistema. Por padrão, todos os tipos de arquivo são permitidos.

Para especificar os tipos de arquivo não listados explicitamente na caixa de diálogo.

1. Selezione outro.
2. Insira um nome para o tipo de arquivo.
3. Insira um tipo de arquivo usando a forma *.suffix - por exemplo, *.xls. É possível especificar diversos tipos de arquivo, cada um separado por um ponto e vírgula.

Guia

O controle de Guia inclui uma guia no diálogo nó. Qualquer um dos outros controles pode ser incluído na nova guia. O controle de Guia tem as propriedades a seguir:

Identificador. O identificador exclusivo para o controle.

Título. O título da guia.

Rankeamento. Especifica a posição da guia no diálogo nó, em relação às demais guias no diálogo nó.

Script. Especifica o script R ou o script Python para Spark que é gerado e executado por esse controle no tempo de execução e pode ser inserido no modelo de script.

- É possível especificar qualquer script R ou script Python para Spark válido e é possível usar \n para quebras de linha.
- O valor %%ThisValue%% gera uma lista separada por espaços em branco do script R ou do script Python para Spark gerado por cada controle na guia, na ordem em que eles aparecem na guia (da parte superior para a parte inferior e da esquerda para a direita). Este é o padrão.
- Se a propriedade Script incluir %%ThisValue%% e nenhum script R ou script Python para Spark for gerado por qualquer um dos controles na guia, então, a guia como um todo não gerará nenhum script.

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Botão de subdiálogo

O controle Botão de subdiálogo especifica um botão para ativação de um subdiálogo e fornece acesso ao Construtor de Diálogo para o subdiálogo. O Botão de subdiálogo possui as propriedades a seguir:

Identificador. O identificador exclusivo para o controle.

TÍTULO. O texto é exibido no botão.

ToolTip. O texto da Dica de ferramenta opcional que aparece quando o usuário passa o mouse sobre o controle.

Subdiálogo. Clique no botão de reticências (...) para abrir o Construtor de diálogo personalizado para o subdiálogo. Também é possível abrir o construtor clicando duas vezes no botão de subdiálogo.

Tecla Mnemônica. Um caractere opcional no título a ser usado como um atalho de teclado para o controle. O caractere aparece sublinhado no título. O atalho é ativado ao pressionar Alt+[tecla mnemônica].

Ativando regra. Especifica uma regra que determina quando o controle atual é ativado. Clique no botão de reticências (...) para abrir a caixa de diálogo e especifique a regra. A propriedade Regra de Ativação fica visível somente quando outros controles que podem ser usados para especificar uma regra de ativação existem na tela.

Nota: O controle Botão de subdiálogo não pode ser incluído em um subdiálogo.

Propriedades de diálogo para um subdiálogo

Para visualizar e configurar propriedades para um subdiálogo:

1. Abra o subdiálogo clicando duas vezes no botão do subdiálogo no diálogo principal, ou dê um clique no botão de subdiálogo e clique no botão de reticências (...) para a propriedade Subdiálogo.
2. No subdiálogo, clique na tela em uma área fora de qualquer controle. Sem controle na tela, as propriedades de um subdiálogo sempre serão visíveis.

Nome do subdiálogo. O identificador exclusivo para o subdiálogo. A propriedade Nome do Subdiálogo é necessária.

Nota: Se você especificar o nome da subcaixa de diálogo como um identificador no modelo 'Script - como em '%My Sub-dialog Name%' - ele será substituído em tempo de execução por uma lista separada por espaços em branco do 'script gerado por cada controle na subcaixa de diálogo, na ordem em que aparecem (de cima para baixo e da esquerda para a direita).

Título. Especifica o texto a ser exibido na barra de título da caixa de subdiálogo. A propriedade Título é opcional, mas recomendado.

Arquivo de ajuda. Especifica o caminho para um arquivo de ajuda opcional para o subdiálogo. Este é o arquivo que será ativado quando o usuário clicar no botão **Ajuda** no subdiálogo e pode ser o mesmo arquivo de ajuda especificado para o diálogo principal. Arquivos de ajuda devem estar no formato HTML. Consulte a descrição da propriedade Arquivo de Ajuda para [Propriedades do diálogo](#) para obter mais informações.

Especificando uma Regra de Ativação para um Controle

É possível especificar uma regra que determine quando um controle é ativado. Por exemplo, é possível especificar que um grupo de opções seja ativado quando um lista de metas é preenchido. As opções disponíveis para especificar a regra de ativação dependem do tipo de controle que define a regra.

Seletor de Campo

É possível especificar que o controle atual é ativado quando uma Seletor de Campo é preenchido com pelo menos um campo (Não vazio). Como alternativa, é possível especificar que o controle atual é ativado quando uma Seletor de Campo não é preenchido (Vazio).

Caixa de Seleção ou Grupo de Caixas de Seleção

É possível especificar que o controle atual é ativado quando uma Caixa de Seleção ou um Grupo de Caixas de Seleção é marcado. Como alternativa, é possível especificar que o controle atual é ativado quando uma Caixa de Seleção ou um Grupo de Caixas de Seleção não é marcado.

Caixa de Combinação ou Caixa de Listagem de Seleção Única

É possível especificar que o controle atual é ativado quando um determinado valor é selecionado em uma Caixa de Combinação ou Caixa de Listagem de Seleção Única. Como alternativa, é possível especificar que o controle atual é ativado quando um determinado valor não é selecionado em uma Caixa de Combinação ou Caixa de Listagem de Seleção Única.

Caixa de Listagem Multisseleção

É possível especificar que o controle atual é ativado quando um determinado valor estiver entre os valores selecionados em uma Caixa de Listagem Multisseleção. Como alternativa, é possível especificar que o controle atual é ativado quando um determinado valor não estiver entre os valores selecionados em uma Caixa de Listagem Multisseleção.

Grupo de opções

É possível especificar que o controle atual é ativado quando um determinado botão de opções é selecionado. Como alternativa, é possível especificar que o controle atual é ativado quando um determinado botão de opções não é selecionado.

Os controles para os quais uma regra de ativação pode ser especificada possuem uma propriedade Regra de Ativação associada.

Nota:

- As regras de ativação são aplicadas, independentemente se o controle que define a regra estiver ativado ou não. Por exemplo, considere uma regra que especifica que um grupo de opção seja ativado quando um lista de metas for preenchido. O grupo de opção é então ativado sempre que o lista de metas é preenchido, independentemente se o lista de metas está ativado.
- Quando o controle de Guia é desativado, todos os controles na guia são desativados, independentemente se qualquer um desses controles tiver uma regra de ativação que esteja satisfeita.
- Quando um Grupo de Caixa de Seleção é desativado, todos os controles no grupo são desativados, independentemente se a caixa de seleção de controle estiver marcada.

Propriedades de Extensão

O diálogo Propriedades da Extensão especifica informações sobre a extensão atual no Construtor de Diálogo Personalizado para Extensões, como o nome da extensão e os arquivos na extensão.

- Todos os diálogos de nó customizados criados no Construtor de Diálogo Personalizado para Extensão fazem parte de uma extensão.
- Campos na guia Obrigatório do diálogo Propriedades de Extensão devem ser especificados antes de ser possível instalar uma extensão e os diálogos de nó customizados que estão contidos nela.

Para especificar as propriedades para uma extensão, nos menus no Construtor de Diálogo Personalizado para Extensões escolha:

Extensão > Propriedades

Propriedades de extensão obrigatórias

Nome

Um nome exclusivo para associar à extensão. Ele pode consistir em até três palavras e não faz distinção entre maiúsculas e minúsculas. Os caracteres são restritos a ASCII de sete bits. Para minimizar a possibilidade de conflitos de nome, talvez você queira utilizar um nome de múltiplas palavras, em que a primeira palavra é um identificador para sua organização, como uma URL. Observe que o nome também será usado para o pacote de extensão (.mpe) nome do arquivo por padrão quando você salvar a extensão. Recomendamos usar o nome padrão quando você economizar. Se você economizar com um nome diferente, você não conseguirá desinstalar a extensão no futuro.

Resumo

Uma descrição simples da extensão que deve ser exibida como uma única linha.

Versão

Um identificador de versão no formato x.x.x, em que cada componente do identificador deve ser um número inteiro, como 1.0.0. Zeros serão implicados se não forem fornecidos. Por exemplo, um

identificador de versão de 3.1 implica 3.1.0. O identificador de versão é independente da versão do IBM SPSS Modeler.

Versão mínima de ModeladorSPSS

A versão mínima de ModeladorSPSS necessária para executar a extensão.

Arquivos

A lista Arquivos exibe os arquivos atualmente incluídos na extensão. Clique em **Incluir** para incluir arquivos na extensão. Também é possível remover arquivos da extensão e é possível extrair arquivos para uma pasta especificada.

- Diálogos de nó customizado têm um tipo de arquivo de .cfe.
- Arquivos de tradução para componentes da extensão são incluídos nas configurações de Localização na guia Opcional.
- É possível incluir um arquivo leia-me na extensão. Especifique o nome do arquivo como ReadMe.txt. Os usuários podem acessar o arquivo leia-me a partir da caixa de diálogo que exibe os detalhes para a extensão. É possível incluir versões localizadas do arquivo leia-me, especificado como ReadMe_<language identifier>.txt, como em ReadMe_fr.txt para uma versão em francês.

Propriedades de extensão opcionais

Propriedades gerais

Descrição

Uma descrição mais detalhada da extensão do que a fornecida para o campo **Resumo**. Por exemplo, você pode listar os principais recursos disponíveis com a extensão.

Data

Uma data opcional para a versão atual da extensão. Nenhuma formatação é fornecida.

Autor

O autor da extensão. Você pode desejar incluir um endereço de e-mail.

Ligações

Um conjunto de URLs para associar à extensão; por exemplo, a página inicial do autor. O formato deste campo é arbitrário, portanto certifique-se de delimitar várias URLs com espaços, vírgulas ou algum outro delimitador razoável.

Palavras-chave

Um conjunto de palavras-chave com as quais associar a extensão.

Plataforma

Informações sobre qualquer restrição que se aplique ao usar a extensão em plataformas de sistemas operacionais em particular.

Dependências

Versão Máxima do Modelador do SPSS

A versão máxima do IBM SPSS Modeler em que a extensão pode ser executada.

Plug-in de integração para R requerido.

Especifica se o Plug-in de integração para R é requerido.

Se a extensão requer qualquer pacote R do repositório de pacotes CRAN, então insira os nomes desses pacotes no controle Pacotes R Necessários. Os Nomes fazem distinção entre maiúsculas e minúsculas. Para incluir o primeiro pacote, clique em qualquer lugar no controle de Pacotes R Necessários para destacar o campo de entrada. Pressionar **Enter** com o cursor em uma determinada linha criará uma nova linha. Você exclui uma linha selecionando-a e pressionando **Excluir**.

Localização

Diálogos de Nós

É possível incluir versões traduzidas do arquivo de propriedades (especifica todas as sequências que aparecem no diálogo de nó) para um diálogo de nó customizado na extensão. Para incluir traduções para um diálogo de nó em particular, selecione o diálogo, clique em **Incluir Traduções** e selecione a pasta que contém as versões traduzidas. Todos os arquivos traduzidos para um diálogo de nó em particular devem estar na mesma pasta. Para obter instruções sobre como criar arquivos de tradução, consulte o tópico “Criando versões localizadas dos diálogos de nó customizados” na página 59.

Pasta de catálogos de tradução

é possível fornecer versões localizadas dos campos **Resumo** e **Descrição** para a extensão que é exibida quando usuários finais visualizam os detalhes da extensão a partir do Hub da Extensão. O conjunto de todos os arquivos localizados para uma extensão deve estar em uma pasta chamada lang. Navegue até a pasta lang que contém os arquivos localizados e selecione essa pasta.

Para fornecer versões localizadas dos campos **Resumo** e **Descrição**, crie um arquivo denominado <extension_name>_<language-identifier>.properties para cada idioma para o qual uma tradução está sendo fornecida. No tempo de execução, se o arquivo .properties para o idioma da interface com o usuário atual não for localizado, os valores dos campos **Resumo** e **Descrição** especificados nas guias Obrigatório e Opcional são usados.

- <extension_name> é o valor do campo **Nome** para a extensão, com quaisquer espaços substituídos por caracteres sublinhado.
- <language-identifier> é o identificador para um determinado idioma. Os identificadores dos idiomas que são suportados pelo IBM SPSS Modeler são mostrados no que vem em seguida.

Por exemplo, as traduções francesas para uma extensão denominada MYORG MYSTAT são armazenadas no arquivo MYORG_MYSTAT_fr.properties.

O arquivo .properties deve conter as duas linhas a seguir, que especificam o texto localizado para os dois campos:

```
Summary=<localized text for Summary field>
Description=<localized text for Description field>
```

- As palavras-chave Sumarização e Descrição devem estar em inglês, e o texto localizado deve estar na mesma linha que a palavra-chave, sem nenhuma quebra de linha.
- O arquivo deve estar na codificação ISO 8859-1. Caracteres que não podem ser diretamente representados nesta codificação devem ser gravados com escapes de Unicode ("\u").

A pasta lang que contém os ficheiros localizados deve ter uma subpasta denominada <language-identifier> que contém o arquivo .properties localizado para um determinado idioma. Por exemplo, o arquivo .properties francês deve estar na pasta lang/fr.

Identificadores de idioma

de. alemão

en. Inglês

es. Espanhol

fr. Francês

it. FAR

ja. japonês

ko. Coreano

pl. Polonês

pt_BR. Português brasileiro

ru. KOR

zh_CN. Chinês simplificado

zh_TW. Chinês tradicional

Gerenciando diálogos nó customizado

O Construtor de Diálogo Personalizado para Extensões permite que você gerencie diálogos de nós customizados em extensões criadas por você ou outros usuários. Os diálogos do nó customizados devem ser instalados em todas as instâncias do Cliente ModeladorSPSS ou SPSS Modeler Batch em que eles são necessários antes que possam ser usados. Observe que não há nada que precise ser instalado para o Servidor do SPSS Modeler para usar um nó de diálogo customizado no modo de servidor.

Nota: É possível modificar apenas os diálogos de nó que são criados no IBM SPSS Modeler.

Abrindo uma extensão que contém diálogos de nós customizados

Você pode abrir um arquivo de pacote de extensão (.mpe) que contém as especificações para um ou mais diálogos personalizados nó ou você pode abrir uma extensão instalada. É possível modificar qualquer um dos diálogos de aprimorados na extensão e salvá-la ou instalá-la. Instalar a extensão instala os diálogos de nós que estão contidos na extensão. Salvar a extensão salva mudanças que foram feitas em qualquer um dos diálogos de nós na extensão.

Para abrir um arquivo de pacote configurável de extensão, nos menus no Construtor de Diálogo Personalizado para Extensões escolha:

Arquivo > Abrir

Para abrir uma extensão instalada, nos menus no Construtor de Diálogo Personalizado para Extensões escolha:

Arquivo > Abrir instalada

Nota: Se você estiver abrindo uma extensão para modificá-la, escolher **Arquivo > Instalar** a reinstala, substituindo a versão existente. Usar **Editar** no menu de contexto de nó criado usando o Construtor de diálogo personalizado não abrirá o diálogo de nó no Construtor de diálogo personalizado.

Salvando um arquivo de pacote configurável de extensão

Salvar uma extensão aberta no Construtor de Diálogo Personalizado para Extensões também salva os diálogos de nó customizado que estão contidos na extensão. As extensões são salvas para um arquivo de pacote de extensão (.mpe). Recomendamos manter o nome do arquivo padrão, que corresponde ao nome especificado no campo **Nome** do diálogo Propriedades de Extensão.

Nos menus no Construtor de diálogo customizado para extensões, escolha:

Arquivo > Salvar

Instalando uma extensão

Instalar uma extensão aberta no Construtor de Diálogo Personalizado para Extensões também instala os diálogos de nó customizado que estão contidos na extensão. Instalar uma instalar existente substitui a versão existente, o que inclui substituir todos os diálogos de nó customizados na extensão que já foram instalados.

Para instalar a extensão aberta atualmente, nos menus no Construtor de Diálogo Personalizado para Extensão escolha:

Arquivo > Instalação

Por padrão, extensões são instaladas em um local geral que pode ser gravado pelo usuário para seus sistemas operacional. Para obter informações adicionais, consulte o tópico [“Locais de instalação para extensões”](#) na página 30.

Nota: Em um fluxo aberto, as versões existentes dos diálogos do nó contidos na extensão não serão substituídas. Ao abrir um fluxo que contenha um nó de Construtor de Diálogo Personalizado que tenha sido reinstalado, você receberá uma mensagem de aviso.

Desinstalando uma extensão

Nos menus no Construtor de diálogo customizado para extensões, escolha:

Arquivo > Desinstalar

Desinstalar uma extensão desinstala todos os diálogos de nó customizado contidos na extensão. Também é possível desinstalar extensões do Hub da Extensão.

Nota: Para desinstalar com sucesso uma extensão, o pacote de extensão .mpe nome do arquivo deve combinar com o nome especificado no diálogo Propriedades de Extensão. Este é o nome de arquivo padrão. Se você modificou o nome do arquivo, renomeá-lo para combinar com o campo Nome e tente se desinstalar novamente.

Importando um arquivo de pacote de diálogo customizado

É possível importar um arquivo de pacote de diálogo customizado (.cfld) no Construtor de Diálogo Personalizado para Extensões. O arquivo .cfld é convertido em um arquivo .cfe, que é incluído em uma nova extensão.

Nos menus no Construtor de diálogo customizado para extensões, escolha:

Arquivo > Importar

Também é possível incluir arquivos .cfe em uma extensão do diálogo Propriedades de Extensão, que é acessada em **Extensão > Propriedades** no Construtor de Diálogo Personalizado para Extensões.

Incluindo um diálogo de nó customizado em uma extensão

É possível incluir um novo diálogo de customizado aprimorado em uma extensão.

Nos menus no Construtor de diálogo customizado para extensões, escolha:

Extensão > Novo diálogo

Alternando entre vários diálogos de nó customizados em uma extensão

Se a extensão atual contém vários diálogos de nó customizado, é possível alternar entre eles.

Nos menus no Construtor de diálogo customizado para extensões, escolha:

Extensão > Editar diálogo e selecione o diálogo customizado do node com o qual você deseja trabalhar.

Criando uma nova extensão

Quando você cria uma nova extensão no Construtor de Diálogo Personalizado para Extensões, um novo diálogo de nó customizado é incluído na extensão.

Para criar uma nova extensão, nos menus no Construtor de Diálogo Personalizado para Extensões escolha:

Arquivo > Novo

Extensões em SPSS Modeler Batch ou em Serviços de Colaboração e Implementação IBM SPSS

Para usar extensões em uma instalação do SPSS Modeler Batch ou Serviços de Colaboração e Implementação IBM SPSS, assegure-se de que a variável de ambiente *IBM_SPSS_MODELER_EXTENSION_PATH* esteja definida no ambiente de destino, e que ela aponte para o local que contém as extensões. Se os fluxos que contém um nó customizado foram armazenados

para o Repositório de serviços de colaboração e implantação IBM SPSS antes que a variável de ambiente *IBM_SPSS_MODELER_EXTENSION_PATH* fosse definida, deve-se rearmazenar os fluxos no repositório antes que eles sejam executados com sucesso.

Nota: Assegure-se de que a versão do adaptador do SPSS Modeler Batch ou Serviços de Colaboração e Implementação IBM SPSS para o Modelador SPSS corresponda à versão do Modelador SPSS Client em que a extensão foi criada.

Criando versões localizadas dos diálogos de nó customizados

É possível criar versões localizadas dos diálogos de nó customizado para qualquer um dos idiomas suportados pelo IBM SPSS Modeler. É possível localizar qualquer sequência que apareça em um diálogo de nó customizado e é possível localizar o nó customizado opcional.

Para localizar sequências de diálogo

Você deve criar uma cópia do arquivo de propriedades associado ao diálogo de nó customizado para cada idioma que planeje implementar. O arquivo de propriedades contém todas as sequências localizáveis associadas ao diálogo de nó.

Extraia o arquivo de diálogo de nó customizado (.cfe) da extensão selecionando o arquivo no diálogo Propriedades de Extensão (no Construtor de Diálogo Personalizado para Extensões) e clicando em **Extrair**. Em seguida, extraia os conteúdos do arquivo .cfe. Um arquivo .cfe é simplesmente um arquivo .zip. O conteúdo extraído de um arquivo .cfe inclui um arquivo de propriedades para cada idioma suportado, em que o nome do arquivo para um determinado idioma é dado por <Dialog Name>_<language identifier>.properties (consulte identificadores de idioma na tabela que segue).

1. Abra cada arquivo de propriedades que você planeja traduzir com um editor de texto que suporte UTF-8, como o Bloco de Notas no Windows. Modifique os valores associados a qualquer propriedade que precisa ser localizada, mas não modifique os nomes das propriedades. As propriedades associadas a um controle específico são prefixadas com o identificador para o controle. Por exemplo, a propriedade ToolTip para um controle com o identificador *options_button* é *options_button_tooltip_LABEL*. As propriedades do título são simplesmente nomeadas <identifier>_LABEL, como em *options_button_LABEL*.
2. Adicione as versões localizadas dos arquivos de propriedades de volta ao arquivo de diálogo de nó personalizado (.cfe) nas configurações de Localização na guia Opcional da caixa de diálogo Propriedades de extensão. Para obter mais informações, consulte o tópico “[Propriedades de extensão opcionais](#)” na página 55.

Quando o diálogo de nó é ativado, IBM SPSS Modeler procura por um arquivo de propriedades cujo identificador de idioma corresponda ao idioma atual, conforme especificado pelo menu suspenso Idioma na guia Geral na caixa de diálogo Opções. Se nenhum arquivo de tais propriedades for encontrado, o arquivo padrão <Dialog Name>.properties é usado.

Para localizar o arquivo de ajuda

1. Faça uma cópia do arquivo de ajuda associado com o diálogo de nó e localize o texto para o idioma que deseja.
2. Renomeie a cópia para <Help File>_<language identifier>.htm, usando os identificadores de idioma na tabela abaixo. Por exemplo, se o arquivo de ajuda for myhelp.htm e você desejar criar uma versão em alemão do arquivo, o arquivo de ajuda localizado deverá ser nomeado como myhelp_de.htm.

Armazene todas as versões localizadas do arquivo de ajuda no mesmo diretório que a versão não localizada. Quando você incluir o arquivo de ajuda não localizado da propriedade Arquivo de Ajuda de Propriedades de Diálogo, as versões localizadas são automaticamente incluídas no diálogo de nó.

Se houver arquivos complementares, como arquivos de imagem, que também precisem ser localizados, então você deve manualmente modificar os caminhos apropriados no arquivo de ajuda principal para

apontar para as versões localizadas. Arquivos suplementares, incluindo versões localizadas, devem ser adicionados manualmente ao arquivo de diálogo de nó personalizado (. cfe). Consulte a seção anterior intitulada "Para localizar sequências de diálogos" para obter informações sobre como acessar e modificar manualmente arquivos de diálogo de nó customizados.

Quando o diálogo de nó é ativado, IBM SPSS Modeler procura por um arquivo de ajuda cujo identificador de idioma corresponda ao idioma atual, conforme especificado pelo menu suspenso Idioma na guia Geral na caixa de diálogo Opções. Se nenhum arquivo de ajuda for localizado, o arquivo de ajuda especificado para o diálogo de nó (o arquivo especificado na propriedade Arquivo de Ajuda de Propriedades do Diálogo) será usado.

Identificadores de idioma

de. alemão

en. Inglês

es. Espanhol

fr. Francês

it. FAR

ja. japonês

ko. Coreano

pl. Polonês

pt_BR. Português brasileiro

ru. KOR

zh_CN. Chinês simplificado

zh_TW. Chinês tradicional

Nota: o texto nos diálogos de nó customizados e os arquivos de ajuda associados não são limitados aos idiomas suportados pelo IBM SPSS Modeler. Você é livre para gravar o diálogo de nó e o texto de ajuda em qualquer idioma sem criar propriedades específicas do idioma e dos arquivos de ajuda. Todos os usuários do seu diálogo de nó verão, então, o texto neste idioma.

Importando e exportando dados usando Python para Spark

Usando o Custom Dialog Builder for Extensions, você pode criar nós personalizados e escrever Python para scripts de Spark para ler dados de onde quer que sua origem de dados esteja, e escrever dados para qualquer formato de dados suportado pelo Apache Spark.

Por exemplo, um usuário deseja gravar seus dados em um banco de dados. Ele usa o Custom Dialog Builder para Extensões e Python para Spark para criar um nó personalizado de exportação JDBC e, em seguida, executa o modelo para gravar dados em um banco de dados. Para ler dados do banco de dados, ele também pode criar um nó JDBC de importação personalizado. Ele também poderia usar este mesmo método para ler dados em ModeladorSPSS a partir de um arquivo JSON, por exemplo. Em seguida, depois de ler seus dados em ModeladorSPSS, ele pode usar todos os nós ModeladorSPSS disponíveis para trabalhar em seu problema de negócios.

Nota: Se você deseja usar o JDBC com Python para a funcionalidade de importação e exportação do Spark, você deve copiar seu arquivo de driver JDBC para o diretório `as/lib` dentro do seu diretório de instalação IBM SPSS Modeler .

Para import/exportar dados usando Python para Spark

1. Vá em **Extensões > Custom Node Dialog Builder**.
2. Em Properties Properties, selecione **Python para Spark** para o Tipo de Script e selecione **Importar** ou **Exportar** para o Tipo de Nô.

3. Insira outras propriedades conforme desejado, como um Nome de Diálogo.
4. Na seção Script, digite ou cole o seu script Python para Spark para importação ou exportação de dados.
5. Clique em **Instalar** para instalar o script Python para Spark. Novos nós de importação personalizados serão adicionados à paleta de Fontes, e novos nós de exportação personalizados serão adicionados à paleta de Exportação.

Importando e exportando dados usando R

Usando o Custom Dialog Builder for Extensions, você pode criar nós personalizados e escrever scripts R para ler dados de onde quer que sua origem de dados esteja, e escrever dados para qualquer formato de dados suportado por R.

Por exemplo, um usuário deseja gravar seus dados em um banco de dados. Ela usa o Custom Dialog Builder para Extensões e R scripting para criar um nó personalizado de exportação JDBC e, em seguida, executa o modelo para gravar dados em um banco de dados. Para ler dados do banco de dados, ela também pode criar um nó JDBC de importação personalizado. Ela também poderia usar este mesmo método para ler dados em ModeladorSPSS a partir de um arquivo JSON, por exemplo. Em seguida, depois de ler seus dados em ModeladorSPSS, ele pode usar todos os nós ModeladorSPSS disponíveis para trabalhar em seu problema de negócios.

Para import/exportar dados usando R

1. Vá em **Extensões > Custom Node Dialog Builder**.
2. Em Properties Properties, selecione **R** para o Tipo de Script e selecione **Importar** ou **Exportar** para o Tipo de Nó.
3. Insira outras propriedades conforme desejado, como um Nome de Diálogo.
4. Na seção Script, digite ou cole o seu script R para importação ou exportação de dados.
5. Clique em **Instalar** para instalar o script R. Novos nós de importação personalizados serão adicionados à paleta de Fontes, e novos nós de exportação personalizados serão adicionados à paleta de Exportação.

Avisos

Estas informações foram desenvolvidas para os produtos e serviços oferecidos nos EUA. Este material pode estar disponível na IBM em outros idiomas. No entanto, pode ser necessário possuir uma cópia do produto ou da versão do produto no mesmo idioma para acessá-lo.

É possível que a IBM não ofereça os produtos, serviços ou recursos discutidos nesta publicação em outros países. Consulte um representante IBM local para obter informações sobre produtos e serviços disponíveis atualmente em sua área. Qualquer referência a produtos, programas ou serviços IBM não significa que apenas produtos, programas ou serviços IBM possam ser utilizados. Qualquer outro produto, programa ou serviço, funcionalmente equivalente, poderá ser utilizado em substituição daqueles, desde que não infrinja nenhum direito de propriedade intelectual da IBM. Entretanto, a avaliação e verificação da operação de qualquer produto, programa ou serviço não IBM são de responsabilidade do Cliente.

A IBM pode ter patentes ou solicitações de patentes pendentes relativas a assuntos tratados nesta publicação. O fornecimento desta publicação não lhe garante direito algum sobre tais patentes. Pedidos de licença devem ser enviados, por escrito, para:

*Gerência de Relações Comerciais e Industriais da IBM Brasil
IBM Corporation
Botafogo
Rio de Janeiro, RJ
EUA*

Para pedidos de licença relacionados a informações de Conjunto de Caracteres de Byte Duplo (DBCS), entre em contato com o Departamento de Propriedade Intelectual da IBM em seu país ou envie pedidos de licença, por escrito, para:

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

A INTERNATIONAL BUSINESS MACHINES CORPORATION FORNECE ESTA PUBLICAÇÃO "NO ESTADO EM QUE SE ENCONTRA", SEM GARANTIA DE NENHUM TIPO, SEJA EXPRESSA OU IMPLÍCITA, INCLUINDO, MAS NÃO SE LIMITANDO ÀS GARANTIAS IMPLÍCITAS DE MERCADO OU DE ADEQUAÇÃO A UM DETERMINADO PROPÓSITO. Alguns países não permitem a exclusão de garantias expressas ou implícitas em certas transações; portanto, essa disposição pode não se aplicar ao Cliente.

Essas informações podem conter imprecisões técnicas ou erros tipográficos. São feitas alterações periódicas nas informações aqui contidas; tais alterações serão incorporadas em futuras edições desta publicação. A IBM pode, a qualquer momento, aperfeiçoar e/ou alterar os produtos e/ou programas descritos nesta publicação, sem aviso prévio.

Referências nestas informações a Web sites não IBM são fornecidas apenas por conveniência e não representam de forma alguma um endosso a esses websites. Os materiais contidos nestes documentos ou Web sites não fazem parte dos materiais deste produto IBM e a utilização destes documentos ou Web sites é de inteira responsabilidade do Cliente.

A IBM pode usar ou distribuir quaisquer informações fornecidas da forma que julgar apropriada sem incorrer em qualquer obrigação para com o Cliente.

Licenciados deste programa que desejam obter informações sobre este assunto com objetivo de permitir: (i) a troca de informações entre programas criados independentemente e outros programas (incluindo este) e (ii) a utilização mútua das informações trocadas, devem entrar em contato com:

*Gerência de Relações Comerciais e Industriais da IBM Brasil
IBM Corporation*

*Botafogo
Rio de Janeiro, RJ
EUA*

Tais informações podem estar disponíveis, sujeitas a termos e condições apropriadas, incluindo em alguns casos o pagamento de uma taxa.

O programa licenciado descrito nesta publicação e todo o material licenciado disponível são fornecidos pela IBM sob os termos do Contrato com o Cliente IBM, do Contrato Internacional de Licença do Programa IBM ou de qualquer outro contrato equivalente.

Os exemplos de clientes e dados de desempenho citados são apresentados com propósitos meramente ilustrativos. Os resultados reais de desempenho podem variar, dependendo das configurações e condições operacionais específicas.

As informações relativas a produtos não IBM foram obtidas junto aos fornecedores dos respectivos produtos, de seus anúncios publicados ou de outras fontes disponíveis publicamente. A IBM não testou estes produtos e não pode confirmar a precisão de seu desempenho, compatibilidade nem qualquer outra reivindicação relacionada a produtos não IBM. Dúvidas sobre os recursos de produtos não IBM devem ser encaminhadas diretamente a seus fornecedores.

As declarações relacionadas aos objetivos e intenções futuras da IBM estão sujeitas a alterações ou cancelamento sem aviso prévio e representam apenas metas e objetivos.

Estas informações contêm exemplos de dados e relatórios utilizados nas operações diárias de negócios. Para ilustrá-los da forma mais completa possível, os exemplos podem incluir nomes de indivíduos, empresas, marcas e produtos. Todos esses nomes são fictícios e qualquer semelhança com pessoas ou empresas reais é mera coincidência.

Marcas comerciais

IBM, o logotipo IBM e ibm.com são marcas comerciais ou marcas registradas da International Business Machines Corp., registradas em várias jurisdições no mundo todo. Outros nomes de produto e de serviço podem ser marcas registradas da IBM ou de outras empresas. Uma lista atual de marcas registradas IBM está disponível na web em "Informações de copyright e marca registrada" em www.ibm.com/legal/copytrade.shtml.

Adobe, o logotipo Adobe, PostScript e o logotipo PostScript são marcas registradas ou marcas comerciais da Adobe Systems Incorporated nos Estados Unidos e/ou em outros países.

Intel, o logotipo Intel, Intel Inside, o logotipo Intel Inside, Intel Centrino, o logotipo do Intel Centrino, Celeron, Intel Xeon, Intel SpeedStep, Itanium e Pentium são marcas comerciais ou marcas registradas da Intel Corporation ou suas subsidiárias nos Estados Unidos e em outros países.

Linux é uma marca registrada da Linus Torvalds nos Estados Unidos e/ou em outros países.

Microsoft, Windows, Windows NT e o logotipo Windows são marcas comerciais da Microsoft Corporation nos Estados Unidos e/ou em outros países.

UNIX é uma marca registrada do The Open Group nos Estados Unidos e/ou em outros países.

Java e todas as marcas comerciais e logotipos baseados em Java são marcas comerciais ou marcas registradas da Oracle e/ou de suas afiliadas.

Termos e condições para documentação do produto

As permissões para uso destas publicações são concedidas sujeitas aos seguintes termos e condições.

Aplicabilidade

Estes termos e condições estão em adição aos termos de uso para o website da IBM.

Uso pessoal

É possível reproduzir estas publicações para seu uso pessoal não comercial, desde que todos os avisos do proprietário sejam preservados. Você pode não distribuir, exibir ou fazer trabalhos derivados dessas publicações, ou de qualquer parte delas, sem o consentimento expresso da IBM.

Uso Comercial

O Cliente pode reproduzir, distribuir e exibir estas publicações unicamente dentro da empresa, desde que todos os avisos do proprietário sejam preservados. Você pode não fazer trabalhos derivados dessas publicações, ou reproduzir, distribuir ou exibir estas publicações ou qualquer parte delas fora de sua empresa, sem o consentimento expresso da IBM.

Direitos

Exceto quando expressamente concedido nesta permissão, nenhuma outra permissão, licença ou direito é concedido, seja de maneira expressa ou implícita, para as publicações ou quaisquer informações, dados, software ou outras propriedades intelectuais aqui contidas.

A IBM reserva-se o direito de retirar as permissões concedidas aqui sempre que, a seu critério, o uso das publicações seja prejudicial a seus interesses ou, conforme determinado pela IBM, as instruções acima não estejam sendo seguidas corretamente.

O Cliente não pode fazer download, exportar ou re-exportar estas informações, exceto se estiver em conformidade total com todas as leis e regulamentos aplicáveis, incluindo todas as leis e regulamentos de exportação dos Estados Unidos.

A IBM NÃO FAZ QUALQUER TIPO DE GARANTIA QUANTO AO CONTEÚDO DESTAS PUBLICAÇÕES. AS PUBLICAÇÕES SÃO FORNECIDAS "NO ESTADO EM QUE SE ENCONTRAM" E SEM GARANTIA DE NENHUM TIPO, SEJA EXPRESSA OU IMPLÍCITA, INCLUINDO, MAS A ELAS NÃO SE LIMITANDO, AS GARANTIAS IMPLÍCITAS DE COMERCIALIZAÇÃO, NÃO INFRAÇÃO E ADEQUAÇÃO A UM DETERMINADO PROPÓSITO.

Índice remissivo

A

Apache Spark [1](#)
arquivo .par [15](#)
arquivo .pim [15](#)

C

Construtor de diálogo personalizado
arquivo de ajuda [33](#)
ativando regras [53](#)
botão de subdiálogo [52](#)
botões de grupo de opções [49](#)
caixa de combinação [38](#)
caixa de listagem [40](#)
caixa de seleção [38](#)
colunas de controle de tabelas [47](#)
controle de data [43](#)
controle de grupo de itens [48](#)
controle de número [42](#)
controle de tabela [46](#)
controle de texto [41](#)
controle de texto estático [45](#)
Exportação de dados r [61](#)
filtrando listas de campos [38](#)
filtro de tipo de arquivo [52](#)
fonte de campo [38](#)
Grupo de caixa de seleção [50](#)
Grupo de opções [49](#)
Importação de dados r [61](#)
itens da lista da caixa de combinação [40](#)
itens da lista da caixa de listagem [40](#)
localizando diálogos e arquivos de ajuda [59](#)
modelo de script [34](#)
Navegador de arquivo [51](#)
Propriedades de subdiálogo [53](#)
propriedades do diálogo [33](#)
Python para exportação de dados Spark [60](#)
Python para importação de dados Spark [60](#)
regras de layout [34](#)
selecionador de cor [46](#)
seletor de campo [37](#)
tabulação [52](#)
texto assegurado [44](#)
visualização [36](#)
Construtor de diálogo personalizado para Extensões
abrindo extensões contendo diálogos de nó [57](#)
arquivos de pacote configurável de extensão [57](#)
instalando extensões contendo diálogos de nó [57](#)
modificando diálogos de nó em extensões instaladas [57](#)
salvando extensões contendo diálogos de nó [57](#)

E

extensões
detalhes de extensão [30](#)
instalando atualizações nas extensões [28](#)

extensões (*continuação*)

localizando e instalando novas extensões [28](#)
removendo extensões [29](#)
visualizando extensões instaladas [29](#)

F

fluxos de publicação
IBM SPSS Modeler Solution Publisher [15](#)

G

Guia de saída de console
Nugget do modelo de extensão [22](#)
guia saída de gráfico
Nugget do modelo de extensão [22](#)
guia saída de texto
Nugget do modelo de extensão [22](#)
guia Sintaxe
Nugget do modelo de extensão [20](#)

I

IBM SPSS Modeler Extensions [1](#)
IBM SPSS Modeler Solution Publisher [15](#)

N

Navegador de Output de extensão [18](#)
Nó de Exportação de extensão
guia saída de console [15](#)
Nó de importação de extensão
guia saída de console [24](#)
Nó de modelo de extensão
guia opções de modelo [20](#)
guia saída de console [20](#)
guia saída de texto [20](#)
Nó de saída de extensão
guia saída [17](#)
guia saída de console [17](#)
guia sintaxe [16](#)
Nó de transformação de extensão
guia saída de console [23](#)
Nós de Extensão [14](#)
Nós de extensão do IBM SPSS Modeler [14](#)
Nugget do modelo de extensão
Guia de saída de console [22](#)
Guia Opções de modelo [21](#)
guia saída de gráfico [22](#)
guia saída de texto [22](#)
guia Sintaxe [20](#)
sobre [20](#)

P

pacotes configuráveis de extensão

pacotes configuráveis de extensão (*continuação*)
instalando pacotes configuráveis de extensão [30](#)
Python for Spark
ambientes de troca [12](#)
API do [2](#), [5](#), [7–10](#)

R

R
sintaxe permitida [1](#)

S

sintaxe R permitida [1](#)

IBM.[®]