

*IBM SPSS Modeler-Erweiterungen*

**IBM**

**Hinweis**

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die Informationen unter „Bemerkungen“ auf Seite 65 gelesen werden.

**Produktinformation**

Diese Ausgabe bezieht sich auf Version 18, Release 0, Modifikation 0 von IBM SPSS Modeler und alle nachfolgenden Releases und Modifikationen, bis dieser Hinweis in einer Neuausgabe geändert wird.

Diese Veröffentlichung ist eine Übersetzung des Handbuchs  
*IBM SPSS Modeler Nodes Guide*,  
herausgegeben von International Business Machines Corporation, USA

© Copyright International Business Machines Corporation 2017

Informationen, die nur für bestimmte Länder Gültigkeit haben und für Deutschland, Österreich und die Schweiz nicht zutreffen, wurden in dieser Veröffentlichung im Originaltext übernommen.

Möglicherweise sind nicht alle in dieser Übersetzung aufgeführten Produkte in Deutschland angekündigt und verfügbar; vor Entscheidungen empfiehlt sich der Kontakt mit der zuständigen IBM Geschäftsstelle.

Änderung des Textes bleibt vorbehalten.

Herausgegeben von:  
TSC Germany  
Kst. 2877  
Juni 2017

---

# Inhaltsverzeichnis

## Kapitel 1. Unterstützte Sprachen . . . . 1

R . . . . .	1
Python for Spark . . . . .	1
Scripting mit Python for Spark . . . . .	2
Erweiterungsknoten . . . . .	12
Erweiterungsexportknoten . . . . .	12
Erweiterungsausgabeknoten . . . . .	15
Erweiterungsmodellknoten . . . . .	17
Erweiterungsmodellnugget . . . . .	18
Erweiterungstransformationsknoten . . . . .	21
Erweiterungsimportknoten . . . . .	22

## Kapitel 2. Erweiterungen . . . . . 25

Erweiterungshub . . . . .	25
Registerkarte "Untersuchen" . . . . .	26
Registerkarte "Installiert" . . . . .	27
Einstellungen . . . . .	28
Erweiterungsdetails . . . . .	28
Installation lokaler Erweiterungsbundles . . . . .	29
Installationsverzeichnisse für Erweiterungen . . . . .	29
Erforderliche R-Pakete . . . . .	29
Erstellen und Verwalten benutzerdefinierter Knoten . . . . .	30
Benutzerdefiniertes Layout der Dialogfelderstellung . . . . .	31

Erstellen eines benutzerdefinierten Dialogfelds für Knoten . . . . .	31
Dialogfeldeigenschaften . . . . .	32
Anordnung der Steuerelemente im Erstellungsbereich für Dialogfelder . . . . .	32
Erstellen der Scriptvorlage . . . . .	33
Anzeigen der Vorschau eines benutzerdefinierten Dialogfelds für Knoten . . . . .	34
Steuerelementtypen . . . . .	35
Erweiterungseigenschaften . . . . .	55
Verwalten von benutzerdefinierten Dialogfeldern für Knoten . . . . .	58
Erstellen lokalisierter Versionen benutzerdefinierter Dialogfelder für Knoten . . . . .	61
Importieren und Exportieren von Daten mithilfe von Python for Spark . . . . .	63
Importieren und Exportieren von Daten mithilfe von R . . . . .	63

## Bemerkungen . . . . . 65

Marken . . . . .	66
Bedingungen für Produktdokumentation . . . . .	66

## Index . . . . . 69



---

# Kapitel 1. Unterstützte Sprachen

IBM® SPSS Modeler unterstützt R und Apache Spark (über Python). Weitere Informationen finden Sie in den folgenden Abschnitten.

---

## R

IBM SPSS Modeler unterstützt R.

### Zulässige Syntax

- Im Syntaxfeld auf der Registerkarte **Syntax** der verschiedenen Erweiterungsknoten sind nur von R erkannte Anweisungen und Funktionen zulässig.
- Beim Erweiterungstransformationsknoten und beim Erweiterungsmodellnugget durchlaufen die Daten das R-Script (im Stapelmodus). Daher sollten R-Scripts für das Modellscoring und Prozessknoten keine Operationen umfassen, die sich über mehrere Zeilen in den Daten erstrecken oder diese zusammenfassen, wie beispielsweise die Sortierung oder die Aggregation. Diese Einschränkung wird auferlegt, um sicherzustellen, dass Daten in einer Hadoop-Umgebung und bei datenbankinternem Mining aufgeteilt werden können. Für Erweiterungsausgabe- und Erweiterungsmodellerstellungsknoten gilt diese Beschränkung nicht.
- Durch den neu hinzugefügten Datenübertragungsmodus ohne Stapelverarbeitung sowohl im Erweiterungstransformationsknoten als auch im Erweiterungsmodellnugget können Sie Zeilen in den Daten in SPSS Modeler Server über mehrere Zeilen verteilen oder zusammenfassen.
- Alle R-Knoten können als unabhängige, globale R-Umgebungen betrachtet werden. Daher muss die R-Bibliothek in beiden R-Scripts geladen werden, wenn die `library`-Funktionen in den beiden separaten R-Knoten verwendet werden sollen.
- Um den Wert eines R-Objekts anzuzeigen, das in Ihrem R-Script definiert ist, müssen Sie einen Aufruf an die Druckfunktion einschließen. Um beispielsweise den Wert eines R-Objekts anzuzeigen, das den Namen `data` hat, nehmen Sie die folgende Zeile in Ihr R-Script auf:  

```
print(data)
```
- Einen Aufruf der R-Funktion `setwd` können Sie in Ihr R-Script nicht aufnehmen, da IBM SPSS Modeler mithilfe dieser Funktion den Dateipfad der R-Scriptausgabedatei steuert.
- Für die Verwendung in CLEM-Ausdrücken und -Scripts definierte Streamparameter werden nicht erkannt, wenn sie in R-Scripts verwendet werden.

---

## Python for Spark

IBM SPSS Modeler unterstützt Python-Scripts für Apache Spark.

### Anmerkung:

- Python-Knoten sind von der Spark-Umgebung abhängig.
- Python-Scripts müssen die Spark-API verwenden, weil die Daten in Form eines Spark-DataFrames dargestellt werden.
- Ältere in Version 17.1 erstellte Knoten können weiterhin für IBM SPSS Analytic Server ausgeführt werden (die Daten stammen von einem IBM SPSS Analytic Server-Quellenknoten und wurden nicht in IBM SPSS Modeler Server extrahiert). Neue in Version 18.0 oder höher erstellte Python-Knoten und Dialogfelderstellungsknoten können für IBM SPSS Modeler Server ausgeführt werden.
- Stellen Sie beim Installieren von Python sicher, dass alle Benutzer die Berechtigung zum Zugriff auf die Python-Installation haben.

- Wenn Sie die MLLib (Machine Learning Library) verwenden wollen, müssen Sie eine Version von Python installieren, die NumPy enthält. Anschließend müssen Sie IBM SPSS Modeler Server (oder den lokalen Server im IBM SPSS Modeler-Client) konfigurieren, um Ihre Python-Installation zu verwenden. Weitere Informationen finden Sie in „Scripting mit Python for Spark“.

## Scripting mit Python for Spark

IBM SPSS Modeler kann zur Verarbeitung von Daten Python-Skripts mithilfe des Apache Spark-Frameworks ausführen. Diese Dokumentation enthält eine Beschreibung der Python-API für die bereitgestellten Schnittstellen.

Die IBM SPSS Modeler-Installation enthält eine Spark-Verteilung (IBM SPSS Modeler 18.1 enthält beispielsweise Spark 2.1.0). Wenn Sie eine andere Version von Spark bevorzugen, können Sie diese konfigurieren, indem Sie die Parameter `eas_spark_path` und `eas_spark_version` in der Datei `options.cfg` hinzufügen. Mögliche Werte für den Parameter `eas_spark_version` sind `1.x` oder `2.x`. Beispiel:

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

## Voraussetzungen

- Wenn Sie Python/Spark-Skripts für IBM SPSS Analytic Server ausführen wollen, muss eine Verbindung zu Analytic Server bestehen und Analytic Server muss Zugriff auf eine kompatible Installation von Apache Spark haben. Einzelheiten zur Verwendung von Apache Spark als Ausführungsengine finden Sie in der Dokumentation zu IBM SPSS Analytic Server.
- Wenn Sie Python/Spark-Skripts für IBM SPSS Modeler Server (oder den im IBM SPSS Modeler-Client enthaltenen lokalen Server, der Windows 64, Linux64 oder Mac64 erfordert), ausführen wollen, müssen Sie Python nicht mehr installieren und `options.cfg` nicht mehr für die Verwendung Ihrer Python-Installation bearbeiten. Ab Version 18.1 enthält IBM SPSS Modeler eine Python-Verteilung. Wenn Sie jedoch ein bestimmtes Modul benötigen, das nicht in der Standard-Python-Verteilung von IBM SPSS Modeler enthalten ist, können Sie `<Modeler-Installationsverzeichnis>/python` aufrufen und weitere Pakete installieren.

Auch wenn jetzt eine Python-Verteilung zum Lieferumfang von IBM SPSS Modeler gehört, können Sie wie in früheren Releases auch auf Ihre eigene Python-Installation verweisen, indem Sie der Datei `options.cfg` die folgende Option hinzufügen:

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

Beispiel:

```
eas_pyspark_python_path, "C:/Ihre_Python-Installation/python.exe"
```

- Wenn Sie den IBM SPSS Modeler-Client unter Linux verwenden, erstellen Sie einen Ordner zum Speichern der Erweiterungen der Dialogfelderstellung. Fügen Sie der Datei `modelerclient.sh` dann den folgenden Befehl hinzu, wobei `<Ihr_CDB-Ordner>` der von Ihnen erstellte Ordner ist:
 

```
export IBM_SPSS_MODELER_EXTENSION_PATH=<Ihr_CDB-Ordner>
```

## IBM SPSS Analytic Server-Kontextobjekt

Der Ausführungskontext für ein Python/Spark-Skript wird durch ein Analytic Server-Kontextobjekt definiert. Bei der Ausführung für IBM SPSS Modeler Server ist das Kontextobjekt für die eingebettete Version von Analytic Server bestimmt, die in der IBM SPSS Modeler Server-Installation enthalten ist. Das Skript muss Folgendes enthalten, damit das Kontextobjekt abgerufen wird:

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

Über den Analytic Server-Kontext können Sie den Spark-Kontext und den SQL-Kontext abrufen:

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

Informationen zum Spark-Kontext und zum SQL-Kontext finden Sie in der Dokumentation zu Apache Spark.

## Zugriff auf Daten

Die Daten werden zwischen einem Python/Spark-Script und dem Ausführungskontext in Form eines Spark SQL-DataFrames übertragen. Ein Script, das Daten verwertet (d. h. alle Knoten, außer einem Quellenknoten), muss den Datenrahmen aus dem Kontext abrufen:

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

Ein Script, das Daten erzeugt (d. h. alle Knoten, außer einem Endknoten), muss einen Datenrahmen an den Kontext zurückgeben:

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

Mithilfe des SQL-Kontexts können Sie bei Bedarf einen Ausgabedatenrahmen von einem RDD (Resilient Distributed Dataset) erstellen:

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

## Definieren des Datenmodells

Ein Knoten, der Daten erzeugt, muss auch ein Datenmodell definieren, das die unterhalb des Knotens angezeigten Felder beschreibt. In der Spark SQL-Terminologie ist das Datenmodell das Schema.

Ein Python/Spark-Script definiert sein Ausgabedatenmodell in Form eines `pyspark.sql.types.StructType`-Objekts. Ein `StructType`-Objekt beschreibt eine Zeile im Ausgabedatenrahmen und wird aus einer Liste von `StructField`-Objekten erstellt. Jedes `StructField`-Objekt beschreibt ein einzelnes Feld im Ausgabedatenmodell.

Sie können das Datenmodell für die Eingabedaten mithilfe des Attributs `:schema` des Eingabedatenrahmens abrufen:

```
inputSchema = inputData.schema
```

Felder, die unverändert übergeben werden, können vom Eingabedatenmodell in das Ausgabedatenmodell kopiert werden. Im Ausgabedatenmodell neue oder geänderte Felder können mithilfe des Konstruktors `StructField` erstellt werden:

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

Informationen zum Konstruktor finden Sie in der Dokumentation zu Spark.

Sie müssen mindestens den Feldnamen und den entsprechenden Datentyp angeben. Optional können Sie Metadaten angeben, um ein Maß, eine Rolle und eine Beschreibung für das Feld anzugeben (siehe „Datenmetadaten“ auf Seite 7).

## DataModelOnly-Modus

IBM SPSS Modeler muss das Ausgabedatenmodell für einen Knoten kennen, bevor der Knoten ausgeführt wird, damit die nachgeordnete Bearbeitung aktiviert wird. Zum Abrufen des Ausgabedatenmodells für einen Python/Spark-Knoten führt IBM SPSS Modeler das Script in einem speziellen `DataModelOnly`-Modus (Modus nur für Datenmodelle) aus, in dem keine Daten verfügbar sind. Das Script kann diesen Modus mithilfe der Methode `isComputeDataModelOnly` für das Analytic Server-Kontextobjekt angeben.

Das Script für einen Transformationsknoten kann das folgende allgemeine Muster verwenden:

```
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)

```

## Erstellen eines Modells

Ein Knoten, der ein Modell erstellt, muss Inhalt an den Ausführungskontext zurückgeben, der das Modell ausreichend beschreibt, damit der Knoten, der das Modell anwendet, es zu einem späteren Zeitpunkt exakt erneut erstellen kann.

Der Modellinhalt wird als Schlüssel/Wert-Paare definiert, wobei die Bedeutung der Schlüssel und Werte nur den Erstellungs- und Scoring-Knoten bekannt ist und von Modeler nicht interpretiert wird. Optional kann der Knoten einem Wert einen MIME-Typ zuordnen, damit Modeler diese Werte mit bekannten Typen dem Benutzer im Modellnugget anzeigen kann.

Ein Wert kann in diesem Kontext beispielsweise PMML, HTML oder ein Bild sein. So fügen Sie dem Modellinhalt (im Erstellungsscript) einen Wert hinzu:

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

So rufen Sie einen Wert aus dem Modellinhalt (im Score-Script) ab:

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

Wenn ein Modell oder ein Teil eines Modells in einer Datei oder einem Ordner im Dateisystem gespeichert wird, können Sie zur Vereinfachung des Vorgangs den gesamten an dieser Position gespeicherten Inhalt in einem Aufruf bündeln (im Erstellungsscript):

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

Beachten Sie, dass es in diesem Fall keine Option zur Angabe eines MIME-Typs gibt, da das Bundle verschiedene Inhaltstypen enthalten kann.

Wenn Sie ein temporäres Verzeichnis zum Speichern des Inhalts während der Erstellung des Modells benötigen, können Sie ein entsprechendes Verzeichnis aus dem Kontext abrufen:

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

So rufen Sie vorhandenen Inhalt in ein temporäres Verzeichnis im Dateisystem (im Score-Script) ab:

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

## Fehlerbehandlung

Zum Ausgeben von Fehlern können Sie eine Ausnahmebedingung vom Script auslösen und sie dem IBM SPSS Modeler-Benutzer anzeigen. Einige Ausnahmebedingungen sind im Modul `spss.pyspark.exceptions` vordefiniert. Beispiel:

```

from spss.pyspark.exceptions import ASContextException
if ... eine Fehlerbedingung ...:
    raise ASContextException("Nachricht, die dem Benutzer angezeigt werden soll")

```

## Analytic Server-Kontext

Der Kontext bietet Unterstützung für die Analytic Server-Kontextschnittstelle für die Interaktion mit IBM SPSS Analytic Server.



## AnalyticServerContext-Objekte

AnalyticServerContext-Objekte richten die Kontextumgebung ein, die mehrere Schnittstellen für die Interaktion mit IBM SPSS Analytic Server bereitstellt. Eine Anwendung, die diese Kontextinstanz erstellen will, muss hierzu die Schnittstelle `spss.pyspark.runtime.getContext()` verwenden, statt die Schnittstelle direkt zu implementieren.

Gibt die Pyspark Python-Instanz `SparkContext` zurück:

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

Gibt die Pyspark Python-Instanz `SQLContext` zurück:

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

Gibt `True` zurück, um anzugeben, ob die Ausführung nur zur Berechnung des Ausgabedatenmodells durchgeführt wird. Andernfalls wird `False` zurückgegeben:

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

Gibt `True` zurück, wenn das Script in der Spark-Umgebung ausgeführt wird. Derzeit wird immer `True` zurückgegeben:

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

Lädt Eingabedaten aus der vorgeordneten temporären Datei und generiert die Instanz `pyspark.sql.DataFrame`:

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

Gibt eine vom Eingabedatenmodell generierte Instanz `pyspark.sql.StructType` zurück. Gibt `None` zurück, wenn das Eingabedatenmodell nicht vorhanden ist:

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

Serialisiert den Ausgabedatenrahmen in einen Analytic Server-Kontext und gibt den Kontext zurück:

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

Parameter:

- `outDF (DataFrame)` : der Wert des Ausgabedatenrahmens

Ausnahmebedingungen:

- `DataOutputNotSupported` : wenn diese Schnittstelle in der Funktion `pyspark:buildmodel` aufgerufen wird
- `ASContextException` : wenn der Ausgabedatenrahmen `None` ist
- `InconsistentOutputDataModel` : die Feldnamen und Speichertypinformationen, die beiden Objekten gemeinsam sind, sind inkonsistent

Wandelt die Instanz `outSchema StructType` in ein Datenmodell um, serialisiert es in den Analytic Server-Kontext und gibt den Kontext zurück:

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

Parameter:

- `outSchema(StructType)` : das `StructType`-Ausgabeobjekt

Ausnahmebedingungen:

- `ASContextException` : wenn die Ausgabeschemainstanz `None` ist
- `InconsistentOutputDataModel` : die Feldnamen und Speichertypinformationen, die beiden Objekten gemeinsam sind, sind inkonsistent

Speichert die Position der Modellerstellungsausgabe im Analytic Server-Kontext und gibt den Kontext zurück:

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

Der Pfad kann ein Verzeichnispfad sein, der die API `cxt.createTemporaryFolder()` zur Generierung verwendet, wenn alle Elemente unter diesem Verzeichnis als Modellinhalt gepackt sind.

Parameter:

- `key (string)` : Schlüsselzeichenfolgewart
- `path (string)` : Zeichenfolge für den Pfad zum Speichern der Modellerstellungsausgabe
- `mimetype (string, optional)` : der MIME-Typ des Inhalts

Ausnahmebedingungen:

- `ModelOutputNotSupported` : wenn diese API nicht über die Funktion `pyspark:buildmodel` aufgerufen wird
- `KeyError` : wenn das Schlüsselattribut `None` ist oder die Zeichenfolge leer ist

Speichert den Modellerstellungsinhalt, die Metadaten und andere Attribute im Analytic Server-Kontext und gibt den Kontext zurück:

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

Parameter:

- `key (string)` : Schlüsselzeichenfolgewart
- `value (string)` : Metadatenzeichenfolgewart des Modells
- `mimetype (string, optional)` : der MIME-Typ des Inhalts

Ausnahmebedingungen:

- `ModelOutputNotSupported` : wenn diese API nicht über die Funktion `pyspark:buildmodel` aufgerufen wird
- `KeyError` : wenn das Schlüsselattribut `None` ist oder die Zeichenfolge leer ist

Gibt die Position des temporären Ordners zurück, der von Analytic Server verwaltet wird. Dieser kann zum Speichern des Modellinhalts verwendet werden.

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

Ausnahmebedingung:

- `ModelOutputNotSupported` : wenn diese API nicht über die Funktion `pyspark:buildmodel` aufgerufen wird

Gibt die Position des Modells zurück, das mit dem Eingabeschlüssel übereinstimmt:

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

Parameter:

- `key (string)` : Schlüsselzeichenfolgewart

Ausnahmebedingungen:

- `ModelInputNotSupported` : wenn diese API nicht über die Funktion `pyspark:applymodel` aufgerufen wird
- `KeyError` : wenn das Schlüsselattribut `None` ist oder die Zeichenfolge leer ist
- `IncompatibleModelContentType` : wenn der Modellinhaltstyp kein Container ist

Gibt den Modellinhalt, die Metadaten des Modells oder andere Modellattribute zurück, die mit dem Eingabeschlüssel übereinstimmen:

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

Parameter:

- key (string) : Schlüsselzeichenfolgewert

Ausnahmebedingungen:

- `ModelInputNotSupported` : wenn diese API nicht über die Funktion `pyspark:applymodel` aufgerufen wird
- `KeyError` : wenn das Schlüsselattribut `None` ist, die Zeichenfolge leer ist oder der Schlüssel nicht vorhanden ist
- `IncompatibleModelContentType` : wenn der Modellinhaltstyp nicht konsistent ist

Gibt den MIME-Typ zurück, der dem Eingabeschlüssel zugeordnet ist. Gibt `None` zurück, wenn der angegebene Inhalt keinen MIME-Typ hat:

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

Parameter:

- key (string) : Schlüsselzeichenfolgewert

Ausnahmebedingungen:

- `ModelInputNotSupported` : wenn diese API nicht über die Funktion `pyspark:applymodel` aufgerufen wird
- `KeyError` : wenn das Schlüsselattribut `None` ist, die Zeichenfolge leer ist oder der Schlüssel nicht vorhanden ist

## Datenmetadaten

In diesem Abschnitt wird die Einrichtung der Datenmodellattribute auf der Grundlage von `pyspark.sql.StructField` beschrieben.

### **spss.datamodel.Role-Objekte**

Diese Klasse zählt gültige Rollen für jedes Feld in einem Datenmodell auf.

**BOTH:** gibt an, dass dieses Feld ein Antezedens oder Sukzedens sein kann.

**FREQWEIGHT:** gibt an, dass dieses Feld als Häufigkeitsgewichtung verwendet wird. Dies wird dem Benutzer nicht angezeigt.

**INPUT:** gibt an, dass dieses Feld ein Prädiktor oder ein Antezedens ist.

**NONE:** gibt an, dass dieses Feld nicht unmittelbar während der Modellierung verwendet wird.

**TARGET:** gibt an, dass dieses Feld vorhergesagt wird oder ein Sukzedens ist.

**PARTITION:** gibt an, dass dieses Feld zur Identifizierung der Datenpartition verwendet wird.

**RECORDID:** gibt an, dass dieses Feld zur Identifizierung der Datensatz-ID verwendet wird.

**SPLIT:** gibt an, dass dieses Feld zum Aufteilen der Daten verwendet wird.

## **spss.datamodel.Measure-Objekte**

Diese Klasse zählt die Messniveaus für Felder in einem Datenmodell auf.

UNKNOWN: gibt an, dass der Messtyp unbekannt ist.

CONTINUOUS: gibt an, dass der Messtyp stetig ist.

NOMINAL: gibt an, dass der Messtyp nominal ist.

FLAG: gibt an, dass der Feldwert einer von zwei Werten ist.

DISCRETE: gibt an, dass der Feldwert als Wertesammlung interpretiert werden soll.

ORDINAL: gibt an, dass der Messtyp ordinal ist.

TYPELESS: gibt an, dass das Feld einen beliebigen Wert haben kann, der mit seinem Speicher kompatibel ist.

## **pyspark.sql.StructField-Objekte**

Stellt ein Feld in einem StructType-Objekt dar. Ein StructField-Objekt enthält vier Felder:

- name (string): Name eines StructField-Objekts
- dataType (pyspark.sql.DataType): bestimmter Datentyp
- nullable (bool): gibt an, ob die Werte eines StructField-Objekts den Wert None enthalten können
- metadata (dictionary): ein Python-Wörterbuch, das zum Speichern der Optionsattribute verwendet wird

Sie können die Metadatenwörterbuchinstanz verwenden, um das Maß-, Rollen- oder Beschriftungsattribut für das bestimmte Feld zu speichern. Es gibt folgende Schlüsselwörter für diese Attribute:

- measure: das Schlüsselwort für das Attribut measure
- role: das Schlüsselwort für das Attribut role
- displayLabel: das Schlüsselwort für das Attribut label

Beispiel:

```
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)
```

## **Datum, Zeit, Zeitmarke**

Bei Operationen, die Daten des Typs Datum, Zeit oder Zeitmarke verwenden, wird der Wert auf der Grundlage des Werts 1970-01-01:00:00:00 (auf der Basis der UTC-Zeit) in den reellen Wert umgewandelt.

Für das Datum stellt der Wert die Anzahl der Tage auf der Grundlage des Werts 1970-01-01 (auf der Basis der UTC-Zeit) dar.

Für die Zeit stellt der Wert die Anzahl der Sekunden (24 Stunden) dar.

Für die Zeitmarke stellt der Wert die Anzahl der Sekunden auf der Grundlage des Werts 1970-01-01:00:00:00 (auf der Basis der UTC-Zeit) dar.

## **Ausnahmebedingungen**

In diesem Abschnitt werden mögliche Ausnahmebedingungen beschrieben.

### **MetadataException-Objekte**

Eine Unterklasse der Python-Ausnahmebedingung.

Diese Ausnahmebedingung wird ausgelöst, wenn während der Verarbeitung des Metadatenobjekts ein Fehler auftritt.

### **UnsupportedOperationException-Objekte**

Eine Unterklasse der Python-Ausnahmebedingung.

Diese Ausnahmebedingung wird ausgelöst, wenn die jeweilige Operation keine Ausführung zulässt.

### **InconsistentOutputDataModel-Objekte**

Eine Unterklasse der Python-Ausnahmebedingung.

Diese Ausnahmebedingung wird ausgelöst, wenn sowohl `setSparkOutputSchema` als auch `setSparkOutputData` aufgerufen werden, aber die Feldnamen und Speichertypinformationen, die beiden Objekten gemeinsam sind, inkonsistent sind.

### **IncompatibleModelContentType-Objekte**

Eine Unterklasse der Python-Ausnahmebedingung.

Diese Ausnahmebedingung wird während der folgenden Szenarios ausgelöst:

- Wenn `setModelContentFormString` zum Festlegen des Modells, aber `getModelContentToPath` zum Abrufen des Werts verwendet wird
- Wenn `setModelContentFormPath` zum Festlegen des Modells, aber `getModelContentToString` zum Abrufen des Werts verwendet wird

### **DataOutputNotSupported-Objekte**

Eine Unterklasse der Python-Ausnahmebedingung.

Diese Ausnahmebedingung wird in `setSparkOutputData` in einer Ausführung ausgegeben, die von der Funktion `pyspark:buildmodel` verarbeitet wird.

### **ModelInputNotSupported-Objekte**

Eine Unterklasse der Python-Ausnahmebedingung.

Diese Ausnahmebedingung wird nur ausgegeben, wenn das Script nicht die APIs `getModelContentPathByKey` und `getModelContentToString` in der Funktion `pyspark:applymodel` aufruft.

### **ModelOutputNotSupported-Objekte**

Eine Unterklasse der Python-Ausnahmebedingung.

Diese Ausnahmebedingung wird nur ausgegeben, wenn das Script nicht die APIs `setModelContentFromPath` und `setModelContentFromString` in der Funktion `pyspark:buildmodel` aufruft.

## ASContextException-Objekte

Eine Unterklasse der Python-Ausnahmebedingung.

Diese Ausnahmebedingung wird ausgelöst, wenn eine nicht erwartete Laufzeitausnahmebedingung auftritt.

## Beispiele

Dieser Abschnitt enthält Scriptbeispiele für Python for Spark.

### Grundlegendes Scriptbeispiel für die Verarbeitung von Daten

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

cxt = spss.pyspark.runtime.getContext()

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

### Beispielscript für die Modellerstellung mithilfe des Algorithmus "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()

# Feldeinstellungen und Algorithmusparameter

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

# Lineares Regressionsmodell in einem Pfad des Dateisystems speichern

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

# Modelldetails in die Standardausgabe ausgeben

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]))

# Prüfen, ob die erforderlichen Felder in den Eingabedaten enthalten sind

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")

# Zuordnungsfunktion zum Umwandeln von DataFrame-Zeilenobjekten zu MLib LabeledPoint definieren

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))

# DataFrame in ein RDD mit LabeledPoint umwandeln

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

# Modell erstellen

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

# Textbeschreibung des Modells in die Standardausgabe schreiben

dump(model,predictors)

# Modell auf dem Dateisystem und im Inhalt des Ausgabemodells speichern

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

```

## Beispielscript für das Modellscoring mithilfe des Algorithmus "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%']

# Ausgabeschema durch Hinzufügen des Vorhersagefelds berechnen
outputSchema = ascontext.getSparkInputSchema()
outputSchema.fields.append(StructField(prediction_field,
DoubleType(), nullable=True))

# Vorhersage basierend auf einem Regressionsmodell und DataFrame-Zeilenobjekt treffen
# Liste mit den Eingabezeilenwerten und vorhergesagten Werten zurückgeben

```

```

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

# Ein serialisiertes Modell aus dem Dateisystem laden

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

ascontext.setSparkOutputSchema(outputSchema)

if not ascontext.isComputeDataModelOnly():
    # Scoring für die Daten im Eingabe-DataFrame durchführen
    indf = ascontext.getSparkInputData()

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

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

    # Ausgabe-DataFrame mit den Scores erstellen
    sqlCtx = SQLContext(sc)
    outdf = sqlCtx.createDataFrame(scores_rdd,schema=outputSchema)

    # Ausgabe-DataFrame als Ergebnis zurückgeben
    ascontext.setSparkOutputData(outdf)

```

---

## Erweiterungsknoten

Als Ergänzung von IBM SPSS Modeler und den darin enthaltenen Data-Mining-Funktionen bieten die Erweiterungsknoten erfahrenen Benutzern die Möglichkeit, eigene Scripts für R oder Python for Spark zur Datenverarbeitung sowie Modellerstellung und zum Modellscoring einzugeben.

## Erweiterungsexportknoten

Mit dem Erweiterungsexportknoten können Sie Scripts für R oder Python for Spark zum Exportieren von Daten ausführen.

### Erweiterungsexportknoten - Registerkarte "Syntax"

Wählen Sie den Syntaxtyp aus - **R** oder **Python for Spark**. Weitere Informationen finden Sie in den folgenden Abschnitten. Wenn Ihre Syntax bereit ist, können Sie auf **Ausführen** klicken, um den Erweiterungsexportknoten auszuführen.

#### R-Syntax

**R-Syntax.** In diesem Feld können Sie benutzerdefinierte R-Scripterstellungssyntax für die Datenanalyse eingeben oder einfügen.



**Flagfelder umwandeln.** Gibt an, wie Flagfelder behandelt werden. Es gibt zwei Optionen: **Zeichenfolgen in Faktor, ganze und reelle Zahlen in Doppelzeichen** und **Logische Werte ("Wahr", "Falsch")**. Wenn Sie **Logische Werte ("Wahr", "Falsch")** auswählen, gehen die ursprünglichen Werte des Flagfelds verloren. Wenn ein Feld beispielsweise die Werte "Männlich" und "Weiblich" enthält, werden diese in "Wahr" und "Falsch" geändert.

**Fehlende Werte in den R-Wert "nicht verfügbar" (NV) umwandeln.** Bei Auswahl dieser Option werden fehlende Werte in den R-Wert NA umgewandelt. Der Wert NA wird von R verwendet, um fehlende Werte anzugeben. Einige der von Ihnen verwendeten R-Funktionen haben möglicherweise ein Argument, mit dem das Verhalten der Funktion bei Vorhandensein von NA in den Daten gesteuert werden kann. Die Funktion kann beispielsweise das automatische Ausschließen von Datensätzen zulassen, die NA enthalten. Wenn diese Option nicht ausgewählt ist, werden fehlende Werte unverändert an R übergeben und können bei der Ausführung der R-Skripts zu Fehlern führen.

**Datums-/Uhrzeitfelder mit spezieller Steuerung für Zeitzonen in R-Klassen umwandeln.** Bei Auswahl dieser Option werden Variablen mit Datums- oder Datums-/Uhrzeitformaten in Datums-/Uhrzeitobjekte von R umgewandelt. Sie müssen eine der folgenden Optionen auswählen:

- **R POSIXct.** Variablen mit Datums- oder Datums-/Uhrzeitformaten werden in die POSIXct-R-Objekte umgewandelt.
- **R POSIXlt (list).** Variablen mit Datums- oder Datums-/Uhrzeitformaten werden in die POSIXlt-R-Objekte umgewandelt.

**Anmerkung:** Die POSIX-Formate sind erweiterte Optionen. Verwenden Sie diese Optionen nur, wenn Ihr R-Skript für Datums-/Uhrzeitfelder Behandlungsverfahren angibt, für die diese Formate erforderlich sind. Die POSIX-Formate sind nicht auf Variablen mit Zeitformaten anwendbar.

## Python-Syntax

**Python-Syntax.** In diesem Feld können Sie benutzerdefinierte Python-Scrippterstellungssyntax für die Datenanalyse eingeben oder einfügen. Weitere Informationen zu Python for Spark finden Sie in „Python for Spark“ auf Seite 1 und „Scripting mit Python for Spark“ auf Seite 2.

## Erweiterungsexportknoten - Registerkarte "Konsolenausgabe"

Die Registerkarte **Konsolenausgabe** enthält Ausgaben, die empfangen werden, wenn das Skript für R oder Python for Spark auf der Registerkarte **Syntax** ausgeführt wird (z. B. wird bei Verwendung eines R-Skripts die von der R-Konsole empfangene Ausgabe angezeigt, wenn das R-Skript im Feld **R-Syntax** auf der Registerkarte **Syntax** ausgeführt wird). Diese Ausgabe kann Fehlermeldungen oder Warnungen zu R oder Python enthalten, die beim Ausführen des R- oder Python-Skripts erzeugt werden. Die Ausgabe kann in erster Linie für das Debugging des R-Skripts verwendet werden. Die Registerkarte **Konsolenausgabe** enthält außerdem das im Feld **R-Syntax** oder **Python-Syntax** angegebene Skript.

Bei jeder Ausführung des Erweiterungsexportskripts wird der Inhalt der Registerkarte **Konsolenausgabe** durch die von der R-Konsole oder Python for Spark empfangene Ausgabe überschrieben. Die Ausgabe kann nicht bearbeitet werden.

## Veröffentlichen von Streams

Streams werden direkt aus IBM SPSS Modeler mit einem beliebigen Standardexportknoten veröffentlicht: Datenbank-, Flatfile-, Statistics-Export-, Erweiterungsexport-, Datenerfassungsexport-, SAS-Export-, Excel- und XML-Exportknoten. Der Exportknotentyp bestimmt das Format der Ergebnisse, die jedes Mal geschrieben werden, wenn der veröffentlichte Stream mit IBM SPSS Modeler Solution Publisher Runtime oder mit einer externen Anwendung ausgeführt wird. Wenn Sie z. B. Ihre Ergebnisse bei jeder Ausführung des veröffentlichten Streams in eine Datenbank schreiben wollen, verwenden Sie einen Datenbankexportknoten.

## Veröffentlichen eines Streams

1. Öffnen oder erstellen Sie einen Stream auf normale Weise und hängen Sie einen Exportknoten an das Ende an.
2. Geben Sie auf der Registerkarte **Veröffentlichen** im Exportknoten einen Stammmamen für die veröffentlichten Dateien an (d. h. der Dateiname, an den die Erweiterungen `.pim`, `.par` und `.xml` angehängt werden).
3. Klicken Sie auf **Veröffentlichen**, um den Stream zu veröffentlichen, oder wählen Sie **Stream veröffentlichen** aus, um den Stream bei jeder Knotenausführung automatisch zu veröffentlichen.

**Veröffentlichter Name.** Geben Sie den Stammmamen für die veröffentlichten Image- und Parameterdateien an.

- Die **Imagedatei** (`*.pim`) stellt alle Informationen bereit, die Runtime benötigt, um den veröffentlichten Stream so wie zum Zeitpunkt des Exports auszuführen. Wenn Sie sicher sind, dass Sie die Einstellungen für den Stream nicht ändern müssen (z. B. die Eingabedatenquelle oder Ausgabedatendatei), können Sie nur die Imagedatei bereitstellen.
- Die **Parameterdatei** (`*.par`) enthält konfigurierbare Informationen zu Datenquellen, Ausgabedateien und Ausführungsoptionen. Wenn Sie in der Lage sein wollen, die Ein- oder Ausgabe des Streams zu steuern, ohne den Stream erneut zu veröffentlichen, benötigen Sie die Parameterdatei und die Imagedatei.
- Die **Metadatendatei** (`*.xml`) beschreibt die Ein- und Ausgaben des Image und die zugehörigen Datenmodelle. Sie wird von Anwendungen verwendet, die die Laufzeitbibliothek integrieren und die die Struktur der Ein- und Ausgabedaten kennen müssen.

**Anmerkung:** Diese Datei wird nur erzeugt, wenn Sie die Option **Metadaten veröffentlichen** auswählen.

**Parameter veröffentlichen.** Bei Bedarf können Sie Streamparameter in die `*.par`-Datei einschließen. Sie können diese Streamparameterwerte ändern, wenn Sie das Image durch Bearbeitung der `*.par`-Datei oder über die Laufzeit-API ausführen.

Diese Option aktiviert die Schaltfläche **Parameter**. Das Dialogfeld **Parameter veröffentlichen** wird angezeigt, wenn Sie auf die Schaltfläche klicken.

Wählen Sie die in das veröffentlichte Image einzuschließenden Parameter aus, indem Sie die relevante Option in der Spalte **Veröffentlichen** auswählen.

**Bei Streamausführung.** Gibt an, ob der Stream bei der Knotenausführung automatisch veröffentlicht wird.

- **Daten exportieren.** Führt den Exportknoten auf die Standardweise aus, ohne den Stream zu veröffentlichen. (Der Knoten wird in IBM SPSS Modeler so ausgeführt, als wenn IBM SPSS Modeler Solution Publisher nicht verfügbar wäre.) Wenn Sie diese Option auswählen, wird der Stream nur veröffentlicht, wenn Sie im Dialogfeld des Exportknotens explizit auf **Veröffentlichen** klicken. Alternativ können Sie den aktuellen Stream mit dem Tool **Veröffentlichen** in der Symbolleiste oder durch die Verwendung eines Scripts veröffentlichen.
- **Stream veröffentlichen.** Veröffentlicht den Stream zur Bereitstellung mithilfe von IBM SPSS Modeler Solution Publisher. Wählen Sie diese Option aus, wenn der Stream bei jeder Ausführung automatisch veröffentlicht werden soll.

**Anmerkung:**

- Wenn Sie den veröffentlichten Stream mit neuen oder aktualisierten Daten ausführen wollen, beachten Sie, dass die Reihenfolge der Felder in der Eingabedatei mit der Reihenfolge der Felder in der Eingabedatei des Quellenknotens, die im veröffentlichten Stream angegeben ist, übereinstimmen muss.

- Wenn Sie mit externen Anwendungen veröffentlichen, filtern Sie irrelevante Felder oder benennen Sie Felder um, um Eingabeanforderungen zu entsprechen. Beide Aktionen können durch die Verwendung eines Filterknotens vor dem Exportknoten durchgeführt werden.

## Erweiterungsausgabeknoten

Wenn **Ausgabe auf Bildschirm** auf der Registerkarte **Ausgabe** des Dialogfelds "Erweiterungsausgabeknoten" ausgewählt ist, wird die Bildschirmausgabe in einem Ausgabebrowserfenster angezeigt. Die Ausgabe wird außerdem dem Ausgabemanager hinzugefügt. Das Ausgabebrowserfenster verfügt über ein eigenes Set von Menüs, mit denen Sie die Ausgabe drucken, speichern oder in einem anderen Format exportieren können. Das Menü **Bearbeiten** enthält nur die Option **Kopieren**. Der Ausgabebrowser des Erweiterungsausgabeknotens weist zwei Registerkarten auf: die Registerkarte **Textausgabe**, auf der die Textausgabe angezeigt wird, und die Registerkarte **Diagrammausgabe**, auf der Diagramme und Grafiken angezeigt werden.

Wenn **Ausgabe in Datei** auf der Registerkarte **Ausgabe** des Dialogfelds **Erweiterungsausgabeknoten** ausgewählt ist, wird das Ausgabebrowserfenster bei erfolgreicher Ausführung des Erweiterungsausgabeknotens nicht angezeigt.

### Erweiterungsausgabeknoten - Registerkarte "Syntax"

Wählen Sie den Syntaxtyp aus - **R** oder **Python for Spark**. Weitere Informationen finden Sie in den folgenden Abschnitten. Wenn Ihre Syntax bereit ist, können Sie auf **Ausführen** klicken, um den Erweiterungsausgabeknoten auszuführen. Die Ausgabeobjekte werden dem Ausgabemanager oder der im Feld **Dateiname** auf der Registerkarte **Ausgabe** angegebenen Datei (optional) hinzugefügt.

#### R-Syntax

**R-Syntax.** In diesem Feld können Sie benutzerdefinierte R-Scrippterstellungssyntax für die Datenanalyse eingeben oder einfügen.

**Flagfelder umwandeln.** Gibt an, wie Flagfelder behandelt werden. Es gibt zwei Optionen: **Zeichenfolgen in Faktor, ganze und reelle Zahlen in Doppelzeichen** und **Logische Werte ("Wahr", "Falsch")**. Wenn Sie **Logische Werte ("Wahr", "Falsch")** auswählen, gehen die ursprünglichen Werte des Flagfelds verloren. Wenn ein Feld beispielsweise die Werte "Männlich" und "Weiblich" enthält, werden diese in "Wahr" und "Falsch" geändert.

**Fehlende Werte in den R-Wert "nicht verfügbar" (NV) umwandeln.** Bei Auswahl dieser Option werden fehlende Werte in den R-Wert NA umgewandelt. Der Wert NA wird von R verwendet, um fehlende Werte anzugeben. Einige der von Ihnen verwendeten R-Funktionen haben möglicherweise ein Argument, mit dem das Verhalten der Funktion bei Vorhandensein von NA in den Daten gesteuert werden kann. Die Funktion kann beispielsweise das automatische Ausschließen von Datensätzen zulassen, die NA enthalten. Wenn diese Option nicht ausgewählt ist, werden fehlende Werte unverändert an R übergeben und können bei der Ausführung der R-Scripts zu Fehlern führen.

**Datums-/Uhrzeitfelder mit spezieller Steuerung für Zeitzonen in R-Klassen umwandeln.** Bei Auswahl dieser Option werden Variablen mit Datums- oder Datums-/Uhrzeitformaten in Datums-/Uhrzeitobjekte von R umgewandelt. Sie müssen eine der folgenden Optionen auswählen:

- **R POSIXct.** Variablen mit Datums- oder Datums-/Uhrzeitformaten werden in die POSIXct-R-Objekte umgewandelt.
- **R POSIXlt (list).** Variablen mit Datums- oder Datums-/Uhrzeitformaten werden in die POSIXlt-R-Objekte umgewandelt.

**Anmerkung:** Die POSIX-Formate sind erweiterte Optionen. Verwenden Sie diese Optionen nur, wenn Ihr R-Script für Datums-/Uhrzeitfelder Behandlungsverfahren angibt, für die diese Formate erforderlich sind. Die POSIX-Formate sind nicht auf Variablen mit Zeitformaten anwendbar.

## Python-Syntax

**Python-Syntax.** In diesem Feld können Sie benutzerdefinierte Python-Scripterstellungssyntax für die Datenanalyse eingeben oder einfügen. Weitere Informationen zu Python for Spark finden Sie in „Python for Spark“ auf Seite 1 und „Scripting mit Python for Spark“ auf Seite 2.

### Erweiterungsausgabeknoten - Registerkarte "Konsolenausgabe"

Die Registerkarte **Konsolenausgabe** enthält Ausgaben, die empfangen werden, wenn das Script für R oder Python for Spark auf der Registerkarte **Syntax** ausgeführt wird (z. B. wird bei Verwendung eines R-Scripts die von der R-Konsole empfangene Ausgabe angezeigt, wenn das R-Script im Feld **R-Syntax** auf der Registerkarte **Syntax** ausgeführt wird). Diese Ausgabe kann Fehlnachrichten oder Warnungen zu R oder Python enthalten, die beim Ausführen des R- oder Python-Scripts erzeugt werden. Die Ausgabe kann in erster Linie für das Debugging des R-Scripts verwendet werden. Die Registerkarte **Konsolenausgabe** enthält außerdem das im Feld **R-Syntax** oder **Python-Syntax** angegebene Script.

Bei jeder Ausführung des Erweiterungsausgabeknotens wird der Inhalt der Registerkarte **Konsolenausgabe** durch die von der R-Konsole oder Python for Spark empfangene Ausgabe überschrieben. Die Ausgabe kann nicht bearbeitet werden.

### Erweiterungsausgabeknoten - Registerkarte "Ausgabe"

**Ausgabename.** Gibt den Namen der Ausgabe an, die beim Ausführen des Knotens erzeugt wird. Wenn **Auto** ausgewählt ist, wird der Name der Ausgabe je nach dem Scripttyp automatisch auf "R-Ausgabe" oder "Python-Ausgabe" gesetzt. Optional können Sie auch **Benutzerdefiniert** auswählen und einen anderen Namen angeben.

**Ausgabe auf Bildschirm.** Wählen Sie diese Option aus, um die Ausgabe in einem neuen Fenster zu generieren und anzuzeigen. Die Ausgabe wird außerdem dem Ausgabemanager hinzugefügt.

**Ausgabe in Datei.** Wählen Sie diese Option aus, um die Ausgabe in einer Datei zu speichern. Hierdurch werden die Optionsfelder **Ausgabediagramm** und **Ausgabedatei** aktiviert.

**Ausgabediagramm.** Nur aktiviert, wenn **Ausgabe in Datei** ausgewählt wurde. Wählen Sie diese Option aus, um Diagramme zu speichern, die durch das Ausführen des Erweiterungsausgabeknotens in einer Datei generiert wurden. Geben Sie im Feld **Dateiname** einen Dateinamen an, der für die generierte Ausgabe verwendet wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um eine bestimmte Datei und eine bestimmte Position anzugeben. Geben Sie den Dateityp in der Dropdown-Liste **Dateityp** an. Die folgenden Dateitypen sind verfügbar:

- Ausgabeobjekt (.cou)
- HTML (.html)

**Ausgabetext.** Nur aktiviert, wenn **Ausgabe in Datei** ausgewählt wurde. Wählen Sie diese Option aus, um Textausgaben zu speichern, die durch das Ausführen des Erweiterungsausgabeknotens in einer Datei generiert wurden. Geben Sie im Feld **Dateiname** einen Dateinamen an, der für die generierte Ausgabe verwendet wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um eine bestimmte Datei und eine bestimmte Position anzugeben. Geben Sie den Dateityp in der Dropdown-Liste **Dateityp** an. Die folgenden Dateitypen sind verfügbar:

- HTML (.html)
- Ausgabeobjekt (.cou)
- Textdokument (.txt)

### Erweiterungsausgabebrowser

Wenn **Ausgabe auf Bildschirm** auf der Registerkarte **Ausgabe** des Dialogfelds "Erweiterungsausgabeknoten" ausgewählt ist, wird die Bildschirmausgabe in einem Ausgabebrowserfenster angezeigt. Die Ausgabe wird außerdem dem Ausgabemanager hinzugefügt. Das Ausgabebrowserfenster verfügt über ein eigenes

Set von Menüs, mit denen Sie die Ausgabe drucken, speichern oder in einem anderen Format exportieren können. Das Menü **Bearbeiten** enthält nur die Option **Kopieren**. Der Ausgabebrowser des Erweiterungsausgabeknotens weist zwei Registerkarten auf:

- Auf der Registerkarte **Textausgabe** wird Textausgabe angezeigt.
- Auf der Registerkarte **Diagrammausgabe** werden Diagramme angezeigt.

Wenn **Ausgabe in Datei** auf der Registerkarte **Ausgabe** des Dialogfelds **Erweiterungsausgabeknoten** anstelle von **Ausgabe auf Bildschirm** ausgewählt ist, wird das Ausgabebrowserfenster bei erfolgreicher Ausführung des Erweiterungsausgabeknotens nicht angezeigt.

**Erweiterungsausgabebrowser - Registerkarte "Textausgabe":** Auf der Registerkarte **Textausgabe** werden Textausgaben angezeigt, die beim Ausführen des Scripts für R oder Python for Spark auf der Registerkarte **Syntax** des Erweiterungsausgabeknotens generiert werden.

**Anmerkung:** Fehlermeldungen oder Warnungen zu R oder Python for Spark, die durch das Ausführen Ihres Erweiterungsausgabescrpts generiert wurden, werden immer auf der Registerkarte **Konsolenausgabe** des Erweiterungsausgabeknotens angezeigt.

**Erweiterungsausgabebrowser - Registerkarte "Diagrammausgabe":** Auf der Registerkarte **Diagrammausgabe** werden Grafiken oder Diagramme angezeigt, die beim Ausführen des Scripts für R oder Python for Spark auf der Registerkarte **Syntax** des Erweiterungsausgabeknotens generiert werden. Enthält Ihr R-Script beispielsweise einen Aufruf der R-Funktion `plot`, wird das erzeugte Diagramm auf dieser Registerkarte angezeigt.

## Erweiterungsmodellknoten

Mit dem Erweiterungsmodellknoten können Sie Scripts für R oder Python for Spark ausführen, um Modelle zu erstellen und für sie ein Scoring durchzuführen.

### Erweiterungsmodellknoten - Registerkarte "Syntax"

Wählen Sie den Syntaxtyp aus - **R** oder **Python for Spark**. Geben Sie dann Ihre angepasste Scripting-Syntax in eins der folgenden Felder ein oder fügen Sie sie ein. Wenn Ihre Syntax bereit ist, können Sie auf **Ausführen** klicken, um den Erweiterungsmodellknoten auszuführen.

#### R-Syntax

**Syntax für R-Modellerstellung.** In diesem Feld können Sie eine benutzerdefinierte R-Scripterstellungssyntax für die Modellerstellung eingeben, einfügen oder anpassen.

**Syntax für R-Modellscoring.** In diesem Feld können Sie eine benutzerdefinierte R-Scripterstellungssyntax für das Modellscoring eingeben, einfügen oder anpassen.

#### Python for Spark-Syntax

**Syntax für Python-Modellerstellung.** In diesem Feld können Sie eine benutzerdefinierte Python-Scripterstellungssyntax für die Modellerstellung eingeben, einfügen oder anpassen.

**Syntax für Python-Modellscoring.** In diesem Feld können Sie eine benutzerdefinierte Python-Scripterstellungssyntax für das Modellscoring eingeben, einfügen oder anpassen.

Weitere Informationen zu Python for Spark finden Sie in „Python for Spark“ auf Seite 1 und „Scripting mit Python for Spark“ auf Seite 2.

## Erweiterungsmodellknoten - Registerkarte "Modelloptionen"

**Modellname.** Sie können den Modellnamen automatisch basierend auf den Ziel- oder ID-Feldnamen (oder dem Modelltyp in Fällen, in denen kein solches Feld angegeben ist) generieren oder einen benutzerdefinierten Namen eingeben.

## Erweiterungsmodellknoten - Registerkarte "Konsolenausgabe"

Die Registerkarte **Konsolenausgabe** enthält Ausgaben, die empfangen werden, wenn das Script für R oder Python for Spark auf der Registerkarte **Syntax** ausgeführt wird (z. B. wird bei Verwendung eines R-Scripts die von der R-Konsole empfangene Ausgabe angezeigt, wenn das R-Script im Feld **R-Syntax** auf der Registerkarte **Syntax** ausgeführt wird). Diese Ausgabe kann Fehlnachrichten oder Warnungen zu R oder Python enthalten, die beim Ausführen des R- oder Python-Scripts erzeugt werden. Die Ausgabe kann in erster Linie für das Debugging des R-Scripts verwendet werden. Die Registerkarte **Konsolenausgabe** enthält außerdem das im Feld **R-Syntax** oder **Python-Syntax** angegebene Script.

Bei jeder Ausführung des Erweiterungsmodellscripts wird der Inhalt der Registerkarte **Konsolenausgabe** durch die von der R-Konsole oder Python for Spark empfangene Ausgabe überschrieben. Die Ausgabe kann nicht bearbeitet werden.

## Erweiterungsmodellknoten - Registerkarte "Textausgabe"

Die Registerkarte **Textausgabe** ist im Erweiterungsmodellknoten verfügbar, wenn das Kontrollkästchen **R-Textausgabe anzeigen** auf der Registerkarte **Modelloptionen** des Dialogfelds für den Erweiterungsmodellknoten ausgewählt wird. Auf dieser Registerkarte kann nur eine Textausgabe angezeigt werden. Auf dieser Registerkarte wird jede Textausgabe angezeigt, die durch das Ausführen des R-Scripts für die Modellerstellung erzeugt wird. Wenn Sie das Script für die Modellerstellung erneut ausführen, ohne zuvor einen anderen Namen für das Modell angegeben zu haben, wird der im Rahmen der vorherigen Ausführung erzeugte Inhalt der Registerkarte **Textausgabe** überschrieben. Die Textausgabe kann nicht bearbeitet werden.

Wenn Sie in Ihrem Script einen Aufruf der R-Funktion `sink` aufnehmen, werden alle Ausgaben, die nach dem Aufruf dieser Funktion erzeugt werden, in der angegebenen Datei gespeichert und nicht auf der Registerkarte **Textausgabe** angezeigt.

**Anmerkung:** Fehlnachrichten oder Warnungen zu R oder Python for Spark, die durch das Ausführen Ihres Scripts für die Modellerstellung generiert wurden, werden immer auf der Registerkarte **Konsolenausgabe** des Erweiterungsmodellknotens angezeigt.

## Erweiterungsmodellnugget

Wenn der Erweiterungsmodellknoten ausgeführt wird, wird ein Erweiterungsmodellnugget generiert und auf der Modellpalette platziert. Dieses Modellnugget enthält das Script für R oder Python for Spark, das die Modellerstellung und das Modellscoring definiert. Standardmäßig enthält das Erweiterungsmodellnugget das für das Modellscoring verwendete Script, Optionen zum Lesen der Daten und, sofern vorhanden, die Ausgabe der R-Konsole oder von Python for Spark. Optional kann das Erweiterungsmodellnugget auch verschiedene andere Modellausgabeformate enthalten wie eine Diagramm- und Textausgabe. Nachdem das Erweiterungsmodellnugget generiert und dem Streamerbereich hinzugefügt wurde, kann ein Ausgabeknoten mit ihm verbunden werden. Der Ausgabeknoten wird dann auf die übliche Weise in IBM SPSS Modeler-Streams verwendet, um Informationen zu Daten und Modellen abzurufen und um Daten in verschiedenen Formaten zu exportieren.

Zur Verwendung dieses Knotens mit R müssen Sie IBM SPSS Modeler - Essentials for R installiert haben. Installationsanweisungen und Informationen zur Kompatibilität finden Sie in *IBM SPSS Modeler - Essentials for R: Installationsanweisungen*. Darüber hinaus muss eine kompatible Version von R auf Ihrem Computer installiert sein.

## Erweiterungsmodellnugget - Registerkarte "Syntax"

Das Erweiterungsmodellnugget weist stets die Registerkarte **Syntax** auf.

**Syntax für R-Modellscoring.** Bei Verwendung von R wird in diesem Feld das R-Script angezeigt, das für das Modellscoring verwendet wird. Dieses Feld ist standardmäßig aktiviert, jedoch nicht bearbeitbar. Klicken Sie zum Bearbeiten des R-Scripts für das Modellscoring auf **Bearbeiten**.

**Syntax für Python-Modellscoring.** Bei Verwendung von Python for Spark wird in diesem Feld das Python-Script angezeigt, das für das Modellscoring verwendet wird. Dieses Feld ist standardmäßig aktiviert, jedoch nicht bearbeitbar. Klicken Sie zum Bearbeiten des Python-Scripts für das Modellscoring auf **Bearbeiten**.

**Bearbeiten.** Klicken Sie auf **Bearbeiten**, damit das Feld mit der Syntax für Scoring bearbeitbar wird. Daraufhin kann das Script für das Modellscoring bearbeitet werden, indem im Feld mit der Syntax für Scoring Eingaben vorgenommen werden. Die Bearbeitung Ihres Scripts für das Modellscoring kann beispielsweise erforderlich sein, wenn Sie nach dem Ausführen des Erweiterungsmodellnuggets einen Fehler in Ihrem Script für das Modellscoring feststellen. Änderungen, die am Script für das Modellscoring im Erweiterungsmodellnugget vorgenommen wurden, gehen verloren, wenn das Modell durch das Ausführen des Erweiterungsmodellknotens neu generiert wird.

## Erweiterungsmodellnugget - Registerkarte "Modelloptionen"

Das Erweiterungsmodellnugget weist stets die Registerkarte **Modelloptionen** auf.

**Datenoptionen lesen.** Diese Optionen gelten nur für R und nicht für Python for Spark. Mit diesen Optionen können Sie angeben, wie fehlende Werte, Flagfelder und Variablen mit Datums- oder Datums-/Uhrzeitformaten behandelt werden sollen.

- **Daten in Stapeln lesen.** Wenn Sie eine große Menge Daten verarbeiten (die z. B. für den Speicher der R-Engine zu groß ist), können Sie die Daten mithilfe dieser Option in Stapel unterteilen, die einzeln gesendet und verarbeitet werden können. Geben Sie die maximale Anzahl Datensätze für die einzelnen Stapel an.

Beim Erweiterungstransformationsknoten sowie beim Erweiterungs-Scoring-Nugget durchlaufen die Daten das R-Script (im Stapelmodus). Daher sollten Scripts für das Modellscoring und Prozessknoten, die in einer Hadoop-Umgebung oder in einer Datenbankumgebung ausgeführt werden, keine Operationen umfassen, die sich über mehrere Zeilen in den Daten erstrecken oder diese zusammenfassen, wie beispielsweise die Sortierung oder die Aggregation. Diese Einschränkung wird auferlegt, um sicherzustellen, dass Daten in einer Hadoop-Umgebung und bei datenbankinternem Mining aufgeteilt werden können. Diese Einschränkung gilt nicht, wenn die Scripts für das Modellscoring in SPSS Modeler Server ausgeführt werden. Für Erweiterungsausgabe- und Erweiterungsmodellknoten gilt diese Beschränkung nicht.

- **Flagfelder umwandeln.** Gibt an, wie Flagfelder behandelt werden. Es gibt zwei Optionen: **Zeichenfolgen in Faktor, ganze und reelle Zahlen in Doppelzeichen** und **Logische Werte ("Wahr", "Falsch")**. Wenn Sie **Logische Werte ("Wahr", "Falsch")** auswählen, gehen die ursprünglichen Werte des Flagfelds verloren. Wenn ein Feld beispielsweise die Werte "Männlich" und "Weiblich" enthält, werden diese in "Wahr" und "Falsch" geändert.
- **Fehlende Werte in den R-Wert "nicht verfügbar" (NV) umwandeln.** Bei Auswahl dieser Option werden fehlende Werte in den R-Wert NA umgewandelt. Der Wert NA wird von R verwendet, um fehlende Werte anzugeben. Einige der von Ihnen verwendeten R-Funktionen haben möglicherweise ein Argument, mit dem das Verhalten der Funktion bei Vorhandensein von NA in den Daten gesteuert werden kann. Die Funktion kann beispielsweise das automatische Ausschließen von Datensätzen zulassen, die NA enthalten. Wenn diese Option nicht ausgewählt ist, werden fehlende Werte unverändert an R übergeben und können bei der Ausführung der R-Scripts zu Fehlern führen.
- **Datums-/Uhrzeitfelder in R-Klassen mit besonderer Zeitzonesteuerung umwandeln.** Bei Auswahl dieser Option werden Variablen mit Datums- oder Datums-/Uhrzeitformaten in Datums-/Uhrzeitobjekte von R umgewandelt. Sie müssen eine der folgenden Optionen auswählen:
  - **R POSIXct.** Variablen mit Datums- oder Datums-/Uhrzeitformaten werden in die POSIXct-R-Objekte umgewandelt.

- **R POSIXlt (list)**. Variablen mit Datums- oder Datums-/Uhrzeitformaten werden in die POSIXlt-R-Objekte umgewandelt.

**Anmerkung:** Die POSIX-Formate sind erweiterte Optionen. Verwenden Sie diese Optionen nur, wenn Ihr R-Script für Datums-/Uhrzeitfelder Behandlungsverfahren angibt, für die diese Formate erforderlich sind. Die POSIX-Formate sind nicht auf Variablen mit Zeitformaten anwendbar.

Die Optionen, die für die Steuerelemente **Flagfelder umwandeln**, **Fehlende Werte in den R-Wert 'nicht verfügbar' (NV)** und **Datums-/Uhrzeitfelder in R-Klassen mit besonderer Zeitzonesteuerung umwandeln** ausgewählt sind, werden nicht erkannt, wenn das Erweiterungsmodellnugget in einer Datenbank ausgeführt wird. Wenn der Knoten in einer Datenbank ausgeführt wird, werden stattdessen die Standardwerte für diese Steuerelemente verwendet:

- **Flagfelder umwandeln** wird auf **Zeichenfolgen in Faktor, ganze und reelle Zahlen in Doppelzeichen** gesetzt.
- **Fehlende Werte in den R-Wert "nicht verfügbar" (NV) umwandeln** wird ausgewählt.
- **Datums-/Uhrzeitfelder in R-Klassen mit besonderer Zeitzonesteuerung umwandeln** wird nicht ausgewählt.

### **Erweiterungsmodellnugget - Registerkarte "Diagrammausgabe"**

Die Registerkarte **Diagrammausgabe** ist im Erweiterungsmodellnugget verfügbar, wenn das Kontrollkästchen **R-Diagramme in HTML anzeigen** auf der Registerkarte **Modelloptionen** des Dialogfelds für den Erweiterungsmodellknoten ausgewählt wird. Auf dieser Registerkarte können Diagramme angezeigt werden, die durch das Ausführen des R-Scripts für die Modellerstellung erzeugt wurden. Enthält Ihr R-Script beispielsweise einen Aufruf der R-Funktion `plot`, wird das erzeugte Diagramm auf dieser Registerkarte angezeigt. Wenn Sie das Script für die Modellerstellung erneut ausführen, ohne zuvor einen anderen Namen für das Modell angegeben zu haben, wird der im Rahmen der vorherigen Ausführung erzeugte Inhalt der Registerkarte **Diagrammausgabe** überschrieben.

### **Erweiterungsmodellnugget - Registerkarte "Textausgabe"**

Die Registerkarte **Textausgabe** ist im Erweiterungsmodellnugget verfügbar, wenn das Kontrollkästchen **R-Textausgabe anzeigen** auf der Registerkarte **Modelloptionen** des Dialogfelds für den Erweiterungsmodellknoten ausgewählt wird. Auf dieser Registerkarte kann nur eine Textausgabe angezeigt werden. Auf dieser Registerkarte wird jede Textausgabe angezeigt, die durch das Ausführen des Erweiterungsmodellscripts erzeugt wird. Wenn Sie das Erweiterungsmodellscript erneut ausführen, ohne zuvor einen anderen Namen für das Modell angegeben zu haben, wird der im Rahmen der vorherigen Ausführung erzeugte Inhalt der Registerkarte **Textausgabe** überschrieben. Die Textausgabe kann nicht bearbeitet werden.

#### **Anmerkung:**

- Wenn Sie in Ihrem Script einen Aufruf der R-Funktion `sink` aufnehmen, werden alle Ausgaben, die nach dem Aufruf dieser Funktion erzeugt werden, in der angegebenen Datei gespeichert und nicht auf der Registerkarte **Textausgabe** angezeigt.
- Fehlernachrichten oder Warnungen, die durch das Ausführen Ihres Erweiterungsmodellscripts generiert wurden, werden stets auf der Registerkarte **Konsolenausgabe** des Erweiterungsmodellknotens angezeigt.

### **Erweiterungsmodellnugget - Registerkarte "Konsolenausgabe"**

Das Erweiterungsmodellnugget weist stets die Registerkarte **Konsolenausgabe** auf. Sie enthält Ausgaben, die empfangen werden, wenn das Script für R oder Python for Spark auf der Registerkarte **Syntax** ausgeführt wird (z. B. wird bei Verwendung eines R-Scripts die von der R-Konsole empfangene Ausgabe angezeigt, wenn das R-Script im Feld **Syntax für R-Modellscoring** auf der Registerkarte **Syntax** des Erweiterungsmodellnuggets ausgeführt wird). Diese Ausgabe enthält alle R- oder Python-Fehlernachrichten oder -Warnungen, die bei der Ausführung des R- oder Python-Scripts generiert werden, sowie alle Textausgaben von der R-Konsole. Die Ausgabe kann in erster Linie für das Debugging des R-Scripts verwendet werden.



Bei jeder Ausführung des Scripts für das Modellscoring wird der Inhalt der Registerkarte **Konsolenausgabe** durch die von der R-Konsole oder Python for Spark empfangenen Ausgabe überschrieben. Die Konsolenausgabe kann nicht bearbeitet werden.

## Erweiterungstransformationsknoten

Über den Erweiterungstransformationsknoten können Sie einem IBM SPSS Modeler-Stream Daten entnehmen und mithilfe von R-Scripting oder Python for Spark-Scripting Transformationen auf Daten anwenden. Wenn die Daten geändert wurden, werden Sie zur weiteren Verarbeitung, Modellerstellung und zum Modellscoring an den Stream zurückgegeben. Über den Erweiterungstransformationsknoten können Daten mithilfe von in R oder Python for Spark geschriebenen Algorithmen transformiert werden. Zudem erhält der Benutzer die Möglichkeit, Methoden zur Datentransformation zu entwickeln, die auf ein spezifisches Problem zugeschnitten sind.

Zur Verwendung dieses Knotens mit R müssen Sie IBM SPSS Modeler - Essentials for R installiert haben. Installationsanweisungen und Informationen zur Kompatibilität finden Sie in *IBM SPSS Modeler - Essentials for R: Installationsanweisungen*. Darüber hinaus muss eine kompatible Version von R auf Ihrem Computer installiert sein.

### Erweiterungstransformationsknoten - Registerkarte "Syntax"

Wählen Sie den Syntaxtyp aus - **R** oder **Python for Spark**. Weitere Informationen finden Sie in den folgenden Abschnitten. Wenn Ihre Syntax bereit ist, können Sie auf **Ausführen** klicken, um den Erweiterungstransformationsknoten auszuführen.

#### R-Syntax

**R-Syntax.** In diesem Feld können Sie benutzerdefinierte R-Scripterstellungssyntax für die Datenanalyse eingeben oder einfügen.

**Flagfelder umwandeln.** Gibt an, wie Flagfelder behandelt werden. Es gibt zwei Optionen: **Zeichenfolgen in Faktor, ganze und reelle Zahlen in Doppelzeichen** und **Logische Werte ("Wahr", "Falsch")**. Wenn Sie **Logische Werte ("Wahr", "Falsch")** auswählen, gehen die ursprünglichen Werte des Flagfelds verloren. Wenn ein Feld beispielsweise die Werte "Männlich" und "Weiblich" enthält, werden diese in "Wahr" und "Falsch" geändert.

**Fehlende Werte in den R-Wert "nicht verfügbar" (NV) umwandeln.** Bei Auswahl dieser Option werden fehlende Werte in den R-Wert NA umgewandelt. Der Wert NA wird von R verwendet, um fehlende Werte anzugeben. Einige der von Ihnen verwendeten R-Funktionen haben möglicherweise ein Argument, mit dem das Verhalten der Funktion bei Vorhandensein von NA in den Daten gesteuert werden kann. Die Funktion kann beispielsweise das automatische Ausschließen von Datensätzen zulassen, die NA enthalten. Wenn diese Option nicht ausgewählt ist, werden fehlende Werte unverändert an R übergeben und können bei der Ausführung der R-Scripts zu Fehlern führen.

**Datums-/Uhrzeitfelder mit spezieller Steuerung für Zeitzonen in R-Klassen umwandeln.** Bei Auswahl dieser Option werden Variablen mit Datums- oder Datums-/Uhrzeitformaten in Datums-/Uhrzeitobjekte von R umgewandelt. Sie müssen eine der folgenden Optionen auswählen:

- **R POSIXct.** Variablen mit Datums- oder Datums-/Uhrzeitformaten werden in die POSIXct-R-Objekte umgewandelt.
- **R POSIXlt (list).** Variablen mit Datums- oder Datums-/Uhrzeitformaten werden in die POSIXlt-R-Objekte umgewandelt.

**Anmerkung:** Die POSIX-Formate sind erweiterte Optionen. Verwenden Sie diese Optionen nur, wenn Ihr R-Script für Datums-/Uhrzeitfelder Behandlungsverfahren angibt, für die diese Formate erforderlich sind. Die POSIX-Formate sind nicht auf Variablen mit Zeitformaten anwendbar.

## Python-Syntax

**Python-Syntax.** In diesem Feld können Sie benutzerdefinierte Python-Scripterstellungssyntax für die Datenanalyse eingeben oder einfügen. Weitere Informationen zu Python for Spark finden Sie in „Python for Spark“ auf Seite 1 und „Scripting mit Python for Spark“ auf Seite 2.

### Erweiterungstransformationsknoten - Registerkarte "Konsolenausgabe"

Die Registerkarte **Konsolenausgabe** enthält Ausgaben, die empfangen werden, wenn das Script für R oder Python for Spark auf der Registerkarte **Syntax** ausgeführt wird (z. B. wird bei Verwendung eines R-Scripts die von der R-Konsole empfangene Ausgabe angezeigt, wenn das R-Script im Feld **R-Syntax** auf der Registerkarte **Syntax** ausgeführt wird). Diese Ausgabe kann Fehlermeldungen oder Warnungen zu R oder Python enthalten, die beim Ausführen des R- oder Python-Scripts erzeugt werden. Die Ausgabe kann in erster Linie für das Debugging des R-Scripts verwendet werden. Die Registerkarte **Konsolenausgabe** enthält außerdem das im Feld **R-Syntax** oder **Python-Syntax** angegebene Script.

Bei jeder Ausführung des Erweiterungstransformationscripts wird der Inhalt der Registerkarte **Konsolenausgabe** durch die von der R-Konsole oder Python for Spark empfangene Ausgabe überschrieben. Die Ausgabe kann nicht bearbeitet werden.

### Erweiterungsimportknoten

Mit dem Erweiterungsimportknoten können Sie Scripts für R oder Python for Spark zum Importieren von Daten ausführen.

#### Erweiterungsimportknoten - Registerkarte "Syntax"

Wählen Sie den Syntaxtyp aus - **R** oder **Python for Spark**. Geben Sie dann Ihr angepasstes Script für das Importieren von Daten oder fügen Sie es ein. Wenn Ihre Syntax bereit ist, können Sie auf **Ausführen** klicken, um den Erweiterungsimportknoten auszuführen.

#### Erweiterungsimportknoten - Registerkarte "Konsolenausgabe"

Die Registerkarte **Konsolenausgabe** enthält Ausgaben, die empfangen werden, wenn das Script für R oder Python for Spark auf der Registerkarte **Syntax** ausgeführt wird (z. B. wird bei Verwendung eines R-Scripts die von der R-Konsole empfangene Ausgabe angezeigt, wenn das R-Script im Feld **R-Syntax** auf der Registerkarte **Syntax** ausgeführt wird). Diese Ausgabe kann Fehlermeldungen oder Warnungen zu R oder Python enthalten, die beim Ausführen des R- oder Python-Scripts erzeugt werden. Die Ausgabe kann in erster Linie für das Debugging des R-Scripts verwendet werden. Die Registerkarte **Konsolenausgabe** enthält außerdem das im Feld **R-Syntax** oder **Python-Syntax** angegebene Script.

Bei jeder Ausführung des Erweiterungsimportscripts wird der Inhalt der Registerkarte **Konsolenausgabe** durch die von der R-Konsole oder Python for Spark empfangene Ausgabe überschrieben. Die Ausgabe kann nicht bearbeitet werden.

### Filtern oder Umbenennen von Feldern

Das Umbenennen und Ausschließen von Feldern ist an jedem beliebigen Punkt in einem Stream möglich. Beispiel: Bei einer medizinischen Studie ist möglicherweise der Kaliumspiegel (Daten der Feldebene) der Patienten (Daten der Datensatzebene) nicht relevant. Daher können Sie das Feld K (Kalium) herausfiltern. Dies ist mithilfe eines gesonderten Filterknotens oder mithilfe der Registerkarte "Filter" in einem Quellen- oder Ausgabeknoten möglich. Die Funktionen sind immer dieselben, unabhängig davon, von welchem Knoten aus der Zugriff erfolgt.

- An Quellenknoten wie beispielsweise "Datei (var.)", "Datei (fest)", "Statistics-Datei", "XML-Datei" oder "Erweiterungsimportdatei" können Sie Felder beim Einlesen der Daten in IBM SPSS Modeler umbenennen oder filtern.
- Mit einem Filterknoten können Sie Felder an jeder Stelle des Streams umbenennen oder filtern.

- Über die Knoten "Statistics-Export", "Statistics-Transformation", "Statistics-Modell" und "Statistics-Ausgabe" können Sie Felder filtern oder umbenennen, die den IBM SPSS Statistics-Benennungsstandards entsprechen. Weitere Informationen finden Sie in Umbenennen oder Filtern von Feldern für IBM SPSS Statistics.
- Mit der Registerkarte "Filter" in einem der oben angegebenen Knoten können Sie Mehrfachantwortsets definieren bzw. bearbeiten. Weitere Informationen finden Sie in Bearbeiten von Mehrfachantwortsets.
- Schließlich können Sie mit einem Filterknoten Felder aus einem Quellenknoten einem anderen Quellenknoten zuweisen.



---

## Kapitel 2. Erweiterungen

Erweiterungen sind angepasste Komponenten, die die Leistungsmerkmale von IBM SPSS Modeler erweitern. Erweiterungen sind in Erweiterungsbundles (.mpe-Dateien) paketi­ert und werden in IBM SPSS Modeler installiert. Erweiterungen können von allen Benutzern erstellt und mit anderen Benutzern gemeinsam genutzt werden, indem das zugehörige Erweiterungsbundle gemeinsam genutzt wird.

Die folgenden Dienstprogramme werden für die Verwendung von Erweiterungen bereitgestellt:

- Der „Erweiterungshub“, auf den über **Erweiterungen > Erweiterungshub** zugegriffen werden kann, ist eine Schnittstelle zum Suchen, Herunterladen und Installieren von Erweiterungen über die IBM SPSS Predictive Analytics-Zusammenstellung für GitHub. Über das Dialogfeld **Erweiterungshub** können Sie auch die Details der Erweiterungen anzeigen, die auf Ihrem Computer installiert sind, Aktualisierungen für installierte Erweiterungen abrufen und Erweiterungen entfernen.
- Sie können ein auf Ihrem lokalen Computer gespeichertes Erweiterungsbundle über **Erweiterungen > Lokales Erweiterungsbundle installieren** installieren.
- Mit der Dialogfelderstellung für Erweiterungen können Sie eine Erweiterung erstellen, die eine Benutzerschnittstelle enthält, die als benutzerdefiniertes Dialogfeld für Knoten bezeichnet wird. Benutzerdefinierte Dialogfelder für Knoten generieren ein R-Script oder ein Python for Spark-Script, das die Aufgaben für die Erweiterung ausführt. Sie können das generierte Script als Teil des Entwurfs des benutzerdefinierten Dialogfelds entwerfen.

---

### Erweiterungshub

Über das Dialogfeld **Erweiterungshub** können Sie folgende Aufgaben ausführen:

- Untersuchen Sie die Erweiterungen, die über die IBM SPSS Predictive Analytics-Zusammenstellung für GitHub verfügbar sind. Sie können Erweiterungen auswählen, die sofort installiert werden sollen, oder ausgewählte Erweiterungen herunterladen und diese zu einem späteren Zeitpunkt installieren.
- Sie können aktualisierte Versionen von Erweiterungen abrufen, die bereits auf Ihrem Computer installiert sind.
- Sie können Details zu den Erweiterungen abrufen, die bereits auf Ihrem Computer installiert sind.
- Sie können Erweiterungen entfernen, die bereits auf Ihrem Computer installiert sind.

So laden Sie Erweiterungen herunter oder entfernen diese:

1. Wählen Sie in den Menüs Folgendes aus: **Erweiterungen > Erweiterungshub**
2. Wählen Sie die Erweiterungen aus, die Sie herunterladen oder entfernen wollen, und klicken Sie dann auf **OK**. Alle Angaben, die Sie auf den Registerkarten **Untersuchen** und **Installiert** machen, werden verarbeitet, wenn Sie auf **OK** klicken.

Standardmäßig werden die von Ihnen für den Download ausgewählten Erweiterungen heruntergeladen und auf Ihrem Computer installiert. Über die Registerkarte **Einstellungen** können Sie auswählen, dass die ausgewählten Erweiterungen an eine angegebene Position heruntergeladen werden, ohne sie zu installieren. Sie können sie dann zu einem späteren Zeitpunkt installieren, indem Sie **Erweiterungen > Lokales Erweiterungsbundle installieren** auswählen. Unter Windows können Sie eine Erweiterung installieren, indem Sie auf die Erweiterungsbundledatei doppelklicken.

#### Wichtig:

- Unter Windows 7 und späteren Windows-Versionen muss IBM SPSS Modeler zum Installieren einer aktualisierten Version eines vorhandenen Erweiterungsbundles möglicherweise mit Administratorrechten ausgeführt werden. Sie können IBM SPSS Modeler mit Administratorrechten starten, indem Sie mit der rechten Maustaste auf das Symbol für IBM SPSS Modeler klicken und **Als Administrator ausführen**

auswählen. Insbesondere, wenn Sie eine Fehlermeldung erhalten, dass mindestens ein Erweiterungsbundle nicht installiert werden konnte, sollten Sie die Installation mit Administratorrechten wiederholen.

- Wenn Sie die Internetverbindung über einen Proxy herstellen und versuchen, den Erweiterungshub durch Auswählen von **Erweiterungen** > **Erweiterungshub** zu öffnen, erhalten Sie möglicherweise einen Fehler, der besagt, dass einige Funktionen möglicherweise nicht verfügbar sind, weil keine Internetverbindung gefunden wurde. Zum Beheben dieses Fehlers fügen Sie den Optionen # JVM in der Datei jvm.cfg (im Verzeichnis 'config' Ihrer Installation von SPSS Modeler) die folgenden Parameter hinzu. Speichern Sie die Datei und starten Sie SPSS Modeler erneut.

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

**Anmerkung:** Sie können die Lizenz, der Sie bei der Installation einer Erweiterung zustimmen, jederzeit später anzeigen, indem Sie für die Erweiterung auf **Weitere Informationen...** auf der Registerkarte **Installiert** klicken.

## Registerkarte "Untersuchen"

Die Registerkarte **Untersuchen** zeigt alle Erweiterungen an, die über die IBM SPSS Predictive Analytics-Zusammenstellung für GitHub (<https://ibmpredictiveanalytics.github.io/>) verfügbar sind. Auf der Registerkarte **Untersuchen** können Sie neue Erweiterungen auswählen, die heruntergeladen und installiert werden sollen, und Sie können Aktualisierungen für Erweiterungen auswählen, die bereits auf Ihrem Computer installiert sind. Für die Registerkarte **Untersuchen** ist eine Internetverbindung erforderlich.

- Für jede Erweiterung wird die Nummer der letzten Version und das entsprechende Datum dieser Version angezeigt. Außerdem wird eine kurze Zusammenfassung der Erweiterung bereitgestellt. Für bereits auf Ihrem Computer installierte Erweiterungen wird zudem die Nummer der installierten Version angezeigt.
- Sie können detailliertere Informationen zu einer Erweiterung anzeigen, indem Sie auf **Weitere Informationen** klicken. Wenn eine Aktualisierung verfügbar ist, werden über **Weitere Informationen** Informationen zu der Aktualisierung angezeigt.
- Sie können die Voraussetzungen zum Ausführen einer Erweiterung anzeigen, beispielsweise ob IBM SPSS Modeler - Integration Plug-in for R erforderlich ist, indem Sie auf **Voraussetzungen** klicken. Wenn eine Aktualisierung verfügbar ist, werden über **Voraussetzungen** Informationen zu der Aktualisierung angezeigt.

## Optimieren nach

Sie können das Set der angezeigten Erweiterungen optimieren. Sie können nach allgemeinen Kategorien von Erweiterungen optimieren, nach der Sprache, in der die Erweiterung implementiert ist, nach dem Typ des Unternehmens, das die Erweiterung bereitgestellt hat oder nach dem Status der Erweiterung. Für jede Gruppe, beispielsweise für "Kategorie", können Sie mehrere Elemente auswählen, nach denen die angezeigte Liste mit Erweiterungen optimiert werden soll. Sie können auch nach Suchbegriffen optimieren. Bei Suchen wird die Groß-/Kleinschreibung nicht beachtet und der Stern (\*) wird als normales Zeichen behandelt und gibt keine Platzhaltersuche an.

- Klicken Sie auf **Anwenden**, um die angezeigte Liste mit Erweiterungen zu optimieren. Wenn Sie die Eingabetaste drücken, während der Cursor im Feld **Suchen** steht, hat dies die gleiche Wirkung wie das Klicken auf **Anwenden**.
- Wenn Sie die Liste zurücksetzen wollen, sodass wieder alle verfügbaren Erweiterungen angezeigt werden, löschen Sie im Feld **Suchen** vorhandenen Text, heben Sie die Auswahl aller Elemente auf und klicken Sie auf **Anwenden**.

## Abrufen von Integrations-Plug-ins

So rufen Sie das IBM SPSS Modeler - Integration Plug-in for R ab:

Installieren Sie IBM SPSS Modeler - Essentials for R, das unter [https://github.com/IBMPredictiveAnalytics/R\\_Essentials\\_Modeler/releases/](https://github.com/IBMPredictiveAnalytics/R_Essentials_Modeler/releases/) oder in der IBM SPSS Predictive Analytics-Community unter <https://developer.ibm.com/predictiveanalytics/predictive-extensions/> verfügbar ist. IBM SPSS Modeler - Essentials for R enthält IBM SPSS Modeler - Integration Plug-in for R. Essentials for R beinhaltet die Programmiersprache R nicht. Vor der Installation von IBM SPSS Modeler - Essentials for R müssen Sie R installieren, wenn es noch nicht installiert ist. Es ist erhältlich unter <http://www.r-project.org/>. R 3.3.3 wird empfohlen.

Detaillierte Informationen zur Installation und Konfiguration von finden Sie im Wiki mit einer Übersicht über IBM SPSS Statistics - Integration Plug-in for R.

**Anmerkung:** Wenn Sie Essentials for R auf einem Computer ohne Internetzugang installieren und die im Lieferumfang von Essentials for R enthaltenen R-Scripts verwenden wollen, müssen Sie alle für diese Scripts erforderlichen R-Pakete anfordern und manuell in R installieren. Wenn Sie die für ein bestimmtes R-Script erforderlichen R-Pakete ermitteln wollen, öffnen Sie das Dialogfeld für den Erweiterungshub (**Erweiterungen** > **Erweiterungshub**), klicken auf die Registerkarte **Installiert** und klicken anschließend auf **Weitere Informationen** für die gewünschte Erweiterung. Die erforderlichen R-Pakete werden im Dialogfeld **Erweiterungsdetails** aufgelistet. R-Pakete können von einer der R-Sites *CRAN Mirrors* heruntergeladen werden, die über <http://www.r-project.org/> aufgerufen werden können. Stellen Sie sicher, dass Sie die Versionen der Pakete herunterladen, die mit Ihrer R-Version übereinstimmen. Die versionsspezifischen Pakete sind über die Links auf der Seite der bereitgestellten Pakete der Site *CRAN Mirrors* verfügbar.

## Registerkarte "Installiert"

Die Registerkarte **Installiert** zeigt alle Erweiterungen an, die auf Ihrem Computer installiert sind. Auf der Registerkarte **Installiert** können Sie Aktualisierungen für installierte Erweiterungen, die über die IBM SPSS Predictive Analytics-Zusammenstellung für GitHub verfügbar sind, auswählen und Erweiterungen entfernen. Zum Abrufen von Aktualisierungen für installierte Erweiterungen ist eine Internetverbindung erforderlich.

- Für jede Erweiterung wird die Nummer der installierten Version angezeigt. Wenn eine Internetverbindung verfügbar ist, wird die Nummer der letzten Version sowie das entsprechende Datum dieser Version angezeigt. Außerdem wird eine kurze Zusammenfassung der Erweiterung bereitgestellt.
- Sie können detailliertere Informationen zu einer Erweiterung anzeigen, indem Sie auf **Weitere Informationen** klicken. Wenn eine Aktualisierung verfügbar ist, werden über **Weitere Informationen** Informationen zu der Aktualisierung angezeigt.
- Sie können die Voraussetzungen zum Ausführen einer Erweiterung anzeigen, beispielsweise ob IBM SPSS Modeler - Integration Plug-in for R erforderlich ist, indem Sie auf **Voraussetzungen** klicken. Wenn eine Aktualisierung verfügbar ist, werden über **Voraussetzungen** Informationen zu der Aktualisierung angezeigt.

## Optimieren nach

Sie können das Set der angezeigten Erweiterungen optimieren. Sie können nach allgemeinen Kategorien von Erweiterungen optimieren, nach der Sprache, in der die Erweiterung implementiert ist, nach dem Typ des Unternehmens, das die Erweiterung bereitgestellt hat oder nach dem Status der Erweiterung. Für jede Gruppe, beispielsweise für "Kategorie", können Sie mehrere Elemente auswählen, nach denen die angezeigte Liste mit Erweiterungen optimiert werden soll. Sie können auch nach Suchbegriffen optimieren. Bei Suchen wird die Groß-/Kleinschreibung nicht beachtet und der Stern (\*) wird als normales Zeichen behandelt und gibt keine Platzhaltersuche an.

- Klicken Sie auf **Anwenden**, um die angezeigte Liste mit Erweiterungen zu optimieren. Wenn Sie die Eingabetaste drücken, während der Cursor im Feld **Suchen** steht, hat dies die gleiche Wirkung wie das Klicken auf **Anwenden**.

- Wenn Sie die Liste zurücksetzen wollen, sodass wieder alle verfügbaren Erweiterungen angezeigt werden, löschen Sie im Feld **Suchen** vorhandenen Text, heben Sie die Auswahl aller Elemente auf und klicken Sie auf **Anwenden**.

## Private Erweiterungen

Private Erweiterungen sind Erweiterungen, die auf Ihrem Computer installiert sind, aber nicht über die IBM SPSS Predictive Analytics-Zusammenstellung für GitHub verfügbar sind. Die Funktionen zur Optimierung des Sets angezeigter Erweiterungen und zur Anzeige der Voraussetzungen für die Ausführung einer Erweiterung sind für private Erweiterungen nicht verfügbar.

**Anmerkung:** Wenn der Erweiterungshub ohne Internetverbindung verwendet wird, sind möglicherweise einige der Funktionen auf der Registerkarte **Installiert** nicht verfügbar.

## Einstellungen

Die Registerkarte **Einstellungen** gibt an, ob für den Download ausgewählte Erweiterungen heruntergeladen und dann auf Ihrem Computer installiert werden oder diese heruntergeladen, aber nicht installiert werden. Diese Einstellung gilt für neue Erweiterungen und Aktualisierungen vorhandener Erweiterungen. Sie könnten zum Beispiel auswählen, dass Erweiterungen zwar heruntergeladen, aber nicht installiert werden sollen, wenn Sie Erweiterungen zur Verteilung an andere Benutzer in Ihrem Unternehmen herunterladen. Oder Sie könnten die Option zum Herunterladen ohne Installieren wählen, wenn die Voraussetzungen zum Ausführen der Erweiterungen derzeit noch nicht erfüllt sind, dies aber für die Zukunft geplant ist.

Wenn Sie Erweiterungen herunterladen, ohne sie zu installieren, können Sie sie zu einem späteren Zeitpunkt installieren, indem Sie **Erweiterungen > Lokales Erweiterungsbundle installieren** auswählen. Unter Windows können Sie eine Erweiterung installieren, indem Sie auf die Erweiterungsbundledatei doppelklicken.

## Erweiterungsdetails

Im Dialogfeld "Erweiterungsdetails" werden die vom Autor der Erweiterung zur Verfügung gestellten Informationen angezeigt. Neben erforderlichen Informationen wie Zusammenfassung und Version hat der Autor unter Umständen URLs zu wichtigen Websites wie der Homepage des Autors eingefügt. Wenn die Erweiterung über den Erweiterungshub heruntergeladen wurde, enthält sie eine Lizenz, die Sie durch Klicken auf **Lizenz anzeigen** anzeigen können.

**Benutzerdefinierte Knoten.** In der Tabelle **Benutzerdefinierte Knoten** werden die benutzerdefinierten Dialogfelder für Knoten aufgelistet, die in der Erweiterung enthalten sind.

**Anmerkung:** Bei der Installation einer Erweiterung, die ein benutzerdefiniertes Dialogfeld für Knoten enthält, kann ein Neustart von IBM SPSS Modeler erforderlich sein, damit der Eintrag für das Dialogfeld für Knoten in der Tabelle Benutzerdefinierte Knoten angezeigt wird.

**Abhängigkeiten.** Die Gruppe **Abhängigkeiten** enthält Add-ons, die zum Ausführen der in der Erweiterung enthaltenen Komponenten erforderlich sind.

- **Integration Plug-In for R.** Für die Komponenten einer Erweiterung ist unter Umständen das Integration Plug-in for R erforderlich.
- **R-Pakete.** Führt alle für die Erweiterung erforderlichen R-Pakete auf. Weitere Informationen finden Sie in „Erforderliche R-Pakete“ auf Seite 29.



---

## Installation lokaler Erweiterungsbundles

So installieren Sie ein auf Ihrem lokalen Computer gespeichertes Erweiterungsbundle:

1. Wählen Sie in den Menüs Folgendes aus:  
**Erweiterungen > Lokales Erweiterungsbundle installieren...**
2. Wählen Sie das Erweiterungsbundle aus. Erweiterungsbundles haben die Dateierweiterung `mpe`.

**Wichtig:** Benutzer von Windows 7 und späteren Windows-Versionen müssen IBM SPSS Modeler zum Installieren einer aktualisierten Version eines vorhandenen Erweiterungsbundles möglicherweise mit Administratorrechten ausführen. Sie können IBM SPSS Modeler mit Administratorrechten starten, indem Sie mit der rechten Maustaste auf das Symbol für IBM SPSS Modeler klicken und **Als Administrator ausführen** auswählen. Insbesondere, wenn Sie eine Fehlermeldung erhalten, dass mindestens ein Erweiterungsbundle nicht installiert werden konnte, sollten Sie die Installation mit Administratorrechten wiederholen.

## Installationsverzeichnisse für Erweiterungen

Standardmäßig werden Erweiterungen an einer allgemeinen Position Ihres Betriebssystems installiert, auf die die Benutzer Schreibzugriff haben.

Sie können die Standardposition überschreiben, indem Sie einen Pfad mit der Umgebungsvariablen `IBM_SPSS_MODELER_EXTENSIONS_PATH` definieren. Die angegebene Position muss auf dem Zielcomputer vorhanden sein. Nachdem Sie den Pfad `IBM_SPSS_MODELER_EXTENSIONS_PATH` festgelegt haben, müssen Sie IBM SPSS Modeler erneut starten, damit die Änderungen wirksam werden.

So erstellen Sie eine Umgebungsvariable über die Systemsteuerung von Windows:

### Windows 7

1. Wählen Sie **Benutzerkonten** aus.
2. Wählen Sie **Umgebungsvariablen ändern** aus.
3. Klicken Sie auf **Neu**, geben Sie in das Feld **Variablenname** den Namen der Umgebungsvariablen (z. B. `IBM_SPSS_MODELER_EXTENSIONS_PATH`) ein und geben Sie den Pfad oder die Pfade in das Feld für den Variablenwert ein.

### Windows 8 oder höher

1. Wählen Sie **System** aus.
2. Wählen Sie die Registerkarte "Erweitert" und klicken Sie auf **Umgebungsvariablen**. Sie gelangen über die erweiterten Systemeinstellungen zur Registerkarte **Erweitert**.
3. Klicken Sie im Abschnitt für Benutzervariablen auf **Neu**, geben Sie in das Feld **Variablenname** den Namen der Umgebungsvariablen (z. B. `IBM_SPSS_MODELER_EXTENSIONS_PATH`) ein und geben Sie den Pfad oder die Pfade in das Feld für den Variablenwert ein.

**Wichtig:** Benutzer von Windows 7 und späteren Windows-Versionen müssen IBM SPSS Modeler zum Installieren einer aktualisierten Version eines vorhandenen Erweiterungsbundles möglicherweise mit Administratorrechten ausführen. Sie können IBM SPSS Modeler mit Administratorrechten starten, indem Sie mit der rechten Maustaste auf das Symbol für IBM SPSS Modeler klicken und **Als Administrator ausführen** auswählen. Insbesondere, wenn Sie eine Fehlermeldung erhalten, dass mindestens ein Erweiterungsbundle nicht installiert werden konnte, sollten Sie die Installation mit Administratorrechten wiederholen.

## Erforderliche R-Pakete

Wenn Sie über keinen Internetzugang verfügen, müssen Sie die erforderlichen R-Pakete für eine bestimmte Erweiterung, die sich nicht auf Ihrem Computer befinden, von einem Computer mit Internetzugang beziehen. Nachdem die Erweiterung installiert wurde, können Sie die Liste der erforderlichen R-Pakete über das Dialogfeld **Erweiterungsdetails** anzeigen. Weitere Informationen finden Sie in

„Erweiterungsdetails“ auf Seite 28. Pakete können unter <http://www.r-project.org/> heruntergeladen und in R installiert werden. Details finden Sie im Handbuch *R Installation and Administration*, das mit R geliefert wird.

**Anmerkung:** Für Benutzer von UNIX (einschließlich Linux) werden die Pakete im Quellenformat heruntergeladen und anschließend zusammengestellt. Dazu müssen die passenden Tools auf Ihrem System installiert sein. Weitere Informationen finden Sie im Handbuch *R Installation and Administration*. Insbesondere Benutzer von Debian sollten das `r-base-dev`-Paket von `apt-get install r-base-dev` installieren.

---

## Erstellen und Verwalten benutzerdefinierter Knoten

Mit der Dialogfelderstellung für Erweiterungen werden Knoten zur Verwendung in SPSS Modeler-Streams erstellt.

Die Dialogfelderstellung für Erweiterungen bietet folgende Möglichkeiten:

- Erstellen Sie ein benutzerdefiniertes Dialogfeld für Knoten, um einen in R oder in Apache Spark (über Python) implementierten Knoten auszuführen. Weitere Informationen finden Sie in „Erstellen der Scriptvorlage“ auf Seite 33.
- Sie können eine Datei mit der Spezifikation für ein benutzerdefiniertes Dialogfeld für Knoten öffnen, das beispielsweise von einem anderen Benutzer erstellt wurde, und das Dialogfeld Ihrer Installation von IBM SPSS Modeler hinzufügen und optional auch Änderungen vornehmen.
- Sie können die Spezifikation für ein benutzerdefiniertes Dialogfeld für Knoten speichern, damit auch andere Benutzer sie ihrer Installation von IBM SPSS Modeler hinzufügen können.
- Sie können benutzerdefinierte Knoten erstellen und Python for Spark-Scripts zum Lesen von Daten aus einer Datenquelle an einer beliebigen Position schreiben sowie Daten in einem beliebigen Format ausgeben, das von Apache Spark unterstützt wird. Weitere Informationen finden Sie in „Importieren und Exportieren von Daten mithilfe von Python for Spark“ auf Seite 63.
- Sie können benutzerdefinierte Knoten erstellen und R-Scripts zum Lesen von Daten aus einer Datenquelle an einer beliebigen Position schreiben sowie Daten in einem beliebigen Format ausgeben, das von R unterstützt wird. Weitere Informationen finden Sie in „Importieren und Exportieren von Daten mithilfe von R“ auf Seite 63.

Mit der Dialogfelderstellung für Erweiterungen können Sie benutzerdefinierte Dialogfelder für Knoten in den Erweiterungen erstellen und ändern. Wenn Sie die Dialogfelderstellung für Erweiterungen öffnen, wird eine neue Erweiterung erstellt, die ein leeres benutzerdefiniertes Dialogfeld für Knoten enthält. Wenn Sie benutzerdefinierte Dialogfelder für Knoten über die Dialogfelderstellung für Erweiterungen speichern oder installieren, werden diese als Teil einer Erweiterung gespeichert bzw. installiert.

### Anmerkung:

- Sie können nicht Ihre eigene Version eines Dialogfelds für Knoten für einen IBM SPSS Modeler-Standardknoten erstellen.
- Das Scripting wird nicht für mit der Dialogfelderstellung erstellte Knoten unterstützt. Hierzu gehören auch mit der Dialogfelderstellung erstellte R-Knoten und Python-Knoten.

## So starten Sie die Dialogfelderstellung für Erweiterungen:

Wählen Sie in den Menüs Folgendes aus: **Erweiterungen > Benutzerdefiniertes Dialogfelderstellungsprogramm**

### Anmerkung:

- Python-Knoten sind von der Spark-Umgebung abhängig.
- Python-Scripts müssen die Spark-API verwenden, weil die Daten in Form eines Spark-DataFrames dargestellt werden.

- Ältere in Version 17.1 erstellte Knoten können weiterhin für IBM SPSS Analytic Server ausgeführt werden (die Daten stammen von einem IBM SPSS Analytic Server-Quellenknoten und wurden nicht in IBM SPSS Modeler Server extrahiert). Neue in Version 18.0 oder höher erstellte Python-Knoten und Dialogfelderstellungsknoten können für IBM SPSS Modeler Server ausgeführt werden.
- Stellen Sie beim Installieren von Python sicher, dass alle Benutzer die Berechtigung zum Zugriff auf die Python-Installation haben.
- Wenn Sie die MLib (Machine Learning Library) verwenden wollen, müssen Sie eine Version von Python installieren, die NumPy enthält. Anschließend müssen Sie IBM SPSS Modeler Server (oder den lokalen Server im IBM SPSS Modeler-Client) konfigurieren, um Ihre Python-Installation zu verwenden. Weitere Informationen finden Sie in „Scripting mit Python for Spark“ auf Seite 2.

## Benutzerdefiniertes Layout der Dialogfelderstellung

### Erstellungsbereich für Dialogfelder

Der Erstellungsbereich für Dialogfelder ist der Bereich der Dialogfelderstellung, in dem Sie das Layout Ihres Dialogfelds für Knoten erstellen.

### Eigenschaftsfenster

Das Eigenschaftsfenster ist der Bereich in der Dialogfelderstellung, in dem Sie die Eigenschaften der Steuerelemente, aus denen das Dialogfeld für Knoten aufgebaut ist, sowie die Eigenschaften des Dialogfelds selbst wie beispielsweise den Knotentyp angeben.

### Toolpalette

Die Toolpalette stellt die Steuerelemente bereit, die in ein benutzerdefiniertes Dialogfeld für Knoten integriert werden können. Sie können die Toolpalette über die Option **Toolpalette** im Menü **Ansicht** auswählen.

### Scriptvorlage

Die Scriptvorlage gibt das R-Script oder Python for Spark-Script an, das vom benutzerdefinierten Dialogfeld für Knoten generiert wird. Sie können das Teilfenster **Scriptvorlage** in ein separates Fenster verschieben, indem Sie auf **In neues Fenster verschieben** klicken. Wenn Sie ein separates Fenster **Scriptvorlage** zurück in die Dialogfelderstellung verschieben wollen, klicken Sie auf **In Hauptfenster wiederherstellen**.

## Erstellen eines benutzerdefinierten Dialogfelds für Knoten

Mit folgenden grundlegenden Schritten erstellen Sie ein benutzerdefiniertes Dialogfeld für Knoten:

1. Legen Sie die Eigenschaften des Dialogfelds für Knoten selbst fest, beispielsweise den Titel, der beim Öffnen des Dialogfelds für Knoten angezeigt wird, und die Position des neuen Knotens in den IBM SPSS Modeler-Paletten. Weitere Informationen finden Sie in „Dialogfeldeigenschaften“ auf Seite 32.
2. Legen Sie die Steuerelemente wie die Feldauswahlen und Kontrollkästchen fest, aus denen das Dialogfeld für Knoten und mögliche Unterdialofelder aufgebaut sind. Weitere Informationen finden Sie in „Steuerelementtypen“ auf Seite 35.
3. Erstellen Sie die Scriptvorlage, die den vom Dialogfeld für Knoten generierten R-Code oder Python for Spark-Code angibt. Weitere Informationen finden Sie in „Erstellen der Scriptvorlage“ auf Seite 33.
4. Geben Sie Eigenschaften der Erweiterung an, die Ihr Dialogfeld für Knoten enthält. Weitere Informationen finden Sie in „Erweiterungseigenschaften“ auf Seite 55.
5. Installieren Sie die Erweiterung, die das Dialogfeld für Knoten enthält, in IBM SPSS Modeler und/oder speichern Sie die Erweiterung in einer Erweiterungsbundledatei (.mpe). Weitere Informationen finden Sie in „Verwalten von benutzerdefinierten Dialogfeldern für Knoten“ auf Seite 58.

Sie können während der Erstellung eine Vorschau des Dialogfelds für Knoten anzeigen. Weitere Informationen finden Sie in „Anzeigen der Vorschau eines benutzerdefinierten Dialogfelds für Knoten“ auf Seite 34.

## Dialogfeldeigenschaften

Das Fenster **Dialogfelderstellung** zeigt die Eigenschaften für das Dialogfeld für Knoten und für das ausgewählte Steuerelement der Benutzerschnittstelle an. Klicken Sie außerhalb der Steuerelemente in den Erstellungsbereich, um die Dialogfeldeigenschaften anzuzeigen und festzulegen. Wenn keine Steuerelemente im Erstellungsbereich enthalten sind, werden die Dialogfeldeigenschaften immer angezeigt.

**Dialogfeldname.** Die Eigenschaft **Dialogfeldname** ist erforderlich und gibt einen eindeutigen Namen an, der dem Dialogfeld für Knoten zugewiesen wird. Um mögliche Namenskonflikte zu vermeiden, können Sie den Namen mit einem Präfix wie dem Namen Ihres Unternehmens oder einem URL versehen.

**Titel.** Die Eigenschaft **Titel** legt den Text fest, der in der Titelleiste des Dialogfelds für Knoten angezeigt wird.

**Hilfdatei.** Die Eigenschaft **Hilfdatei** ist optional und legt den Pfad zur Hilfdatei für das Dialogfeld für Knoten fest. Diese Datei wird gestartet, wenn der Benutzer im Dialogfeld auf die Schaltfläche **Hilfe** klickt. Bei Hilfdateien muss es sich um HTML-Dateien handeln. Eine Kopie der angegebenen Hilfdatei ist in den Spezifikationen für das Dialogfeld für Knoten enthalten, wenn das Dialogfeld für Knoten installiert oder gespeichert wird. Die Hilfeschaltfläche wird im Dialogfeld ausgeblendet, wenn keine Hilfdatei angegeben wird.

- Lokalisierte Versionen der Hilfdatei, die sich in demselben Verzeichnis wie die Hilfdatei befinden, werden dem Dialogfeld für Knoten automatisch hinzugefügt, wenn Sie die Hilfdatei hinzufügen. Der Name lokalisierter Versionen der Hilfdateien lautet `<Hilfdatei>_<Sprach-ID>.htm`. Weitere Informationen finden Sie in „Erstellen lokalisierter Versionen benutzerdefinierter Dialogfelder für Knoten“ auf Seite 61.
- Begleitende Dateien wie Bilddateien und Style-Sheets können dem Dialogfeld für Knoten hinzugefügt werden, indem zuerst das Dialogfeld für Knoten gespeichert wird. Dann fügen Sie diese begleitenden Dateien der Datei (.cfe) mit Dialogfeldern für Knoten manuell hinzu. Informationen zum Zugriff auf Dateien mit benutzerdefinierten Dialogfeldern für Knoten sowie zu ihrer Änderung finden Sie im Abschnitt *Lokalisieren von Zeichenfolgen in Dialogfeldern* im Thema „Erstellen lokalisierter Versionen benutzerdefinierter Dialogfelder für Knoten“ auf Seite 61.

**Scripttyp.** Gibt den Typ von Script an, das zum Erstellen der Scriptvorlage verwendet werden kann. In IBM SPSS Modeler kann R-Scripting oder Python for Spark-Scripting verwendet werden.

**Score aus dem Modell.** Gibt an, ob das Modell, das unter Verwendung des Modellerstellungsscripts erstellt wurde, zum Scoring verwendet werden kann.

**Knotentyp.** Gibt den Typ von Knoten an, der beim Installieren Ihres Dialogfelds für Knoten erstellt wird.

**Palette.** Gibt die Palette an, zu der der neu erstellte Knoten hinzugefügt wird, wenn Sie Ihr Dialogfeld für Knoten installieren.

**Knotensymbol.** Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um ein Bild auszuwählen, das als Knotensymbol für den neu erstellten Knoten verwendet werden soll. Das ausgewählte Bild muss eine GIF-Datei sein.

## Anordnung der Steuerelemente im Erstellungsbereich für Dialogfelder

Sie können Steuerelemente in ein benutzerdefiniertes Dialogfeld für Knoten integrieren, indem Sie sie aus der Toolpalette in den Erstellungsbereich für Dialogfelder ziehen. Damit die Konsistenz mit integrierten

Dialogfeldern für Knoten gewährleistet wird, ist der Erstellungsbereich für Dialogfelder in drei funktionale Spalten unterteilt, in die Sie die Steuerelemente setzen können.

- Die erste Spalte (links außen) ist hauptsächlich für das Steuerelement "Feldauswahl" vorgesehen.
- Schaltflächen für Unterdialogfelder müssen sich in der Spalte rechts außen befinden (beispielsweise in der dritten Spalte, wenn nur drei Spalten verwendet werden) und in derselben Spalte können sich außer den Schaltflächen für Unterdialogfelder keine anderen Steuerelemente befinden. Daher kann die vierte Spalte nur Schaltflächen für Unterdialogfelder enthalten.

Obwohl dies im Erstellungsbereich für Dialogfelder nicht angezeigt wird, werden bei der Installation des Dialogfelds für Knoten in IBM SPSS Modeler die entsprechenden Schaltflächen zum Dialogfeld hinzugefügt (z. B. **OK**, **Abbrechen**, **Anwenden**, **Zurücksetzen** und gegebenenfalls **Hilfe** und **Ausführen**). Die Anzeige und Position dieser Schaltflächen wird automatisch festgelegt. Die Schaltfläche **Hilfe** wird jedoch ausgeblendet, wenn keine Hilfedatei mit dem Dialogfeld für Knoten verknüpft ist (siehe die Eigenschaft **Hilfedatei** in den Dialogfeldeigenschaften).

Sie können die vertikale Reihenfolge der Steuerelemente in einer Spalte ändern, indem Sie sie nach oben oder unten ziehen. Die exakte Position der Steuerelemente wird jedoch automatisch für Sie bestimmt. Während der Ausführung werden die Steuerelemente auf angemessene Art in der Größe geändert, wenn die Größe des Dialogfelds selbst geändert wird. Steuerelemente wie Feldauswahlen werden automatisch an die verfügbare Größe angepasst.

## Erstellen der Scriptvorlage

Die Scriptvorlage gibt das R-Script oder Python for Spark-Script an, das vom benutzerdefinierten Dialogfeld für Knoten generiert wird. Mit einem einzelnen benutzerdefinierten Dialogfeld für Knoten können eine oder mehrere Operationen angegeben werden, die nacheinander ausgeführt werden.

Die Scriptvorlage kann *statischen Text* enthalten. Es besteht ein Unterschied zwischen statischem Text und dem Steuerelement **Statischer Text**. Bei statischem Text handelt es sich um R-Code oder Python for Spark-Code, der bei jeder Ausführung des Knotens generiert wird. Befehlsnamen und Unterbefehlspezifikationen, die von keiner Benutzereingabe abhängig sind, gelten beispielsweise als statischer Text. Die Scriptvorlage kann zudem auch Steuerelement-IDs enthalten, die bei der Ausführung durch die Werte der zugeordneten Steuerelemente für benutzerdefinierten Dialogfelder für Knoten ersetzt werden. Das in einer Feldauswahl angegebene Feldset wird beispielsweise mit der Steuerelement-ID für das Steuerelement "Feldauswahl" dargestellt.

## Erstellen der Scriptvorlage

1. Geben Sie bei statischem Text, der nicht von benutzerdefinierten Werten abhängig ist, das R-Script oder Python for Spark-Script ein, das Sie beispielsweise im Feld **R-Modellerstellungssyntax** des R-Erstellungsknotens eingeben würden.
2. Fügen Sie die Steuerelement-IDs im Format `%ID%` an den Positionen hinzu, an denen von Steuerelementen generierte R-Scripts oder Python for Spark-Scripts eingefügt werden sollen. Hierbei steht ID für den Wert der Eigenschaft "ID" des Steuerelements.
  - Sie können eine Steuerelement-ID einfügen, indem Sie eine Zeile in der Tabelle der IDs auswählen, mit der rechten Maustaste klicken und dann **Zur Scriptvorlage hinzufügen** auswählen. Sie können eine Steuerelement-ID auch einfügen, indem Sie mit der rechten Maustaste auf ein Steuerelement im Erstellungsbereich klicken und **Zur Scriptvorlage hinzufügen** auswählen.
  - Sie können Ihre Auswahl auch aus einer Liste der verfügbaren Steuerelement-IDs treffen, indem Sie Strg+Leertaste drücken. Die Liste enthält die Steuerelement-IDs gefolgt von den Elementen, für die die automatische Scriptvervollständigung zur Verfügung steht.

Wenn Sie IDs manuell eingeben, übernehmen Sie auch Leerzeichen, da sie in den IDs berücksichtigt werden.

Bei der Ausführung und bei allen Steuerelementen außer Kontrollkästchen, Kontrollkästchengruppen und dem Steuerelement "Statischer Text" wird jede ID durch den aktuellen Wert der Eigenschaft

**Script** des zugeordneten Steuerelements ersetzt. Wenn das Steuerelement bei der Ausführung leer ist, generiert es kein Script. Bei Kontrollkästchen und Kontrollkästchengruppen wird die ID, je nach dem aktuellen Status ("Aktiviert" oder "Inaktiviert") des Steuerelements, durch den aktuellen Wert der Eigenschaft "R-Script für Aktivierung" oder "R-Script für Inaktivierung" des zugeordneten Steuerelements ersetzt. Weitere Informationen finden Sie in „Steuerelementtypen“ auf Seite 35.

## Beispiel: Integrieren von Laufzeitwerten in eine R-Script-Vorlage

In diesem Beispiel generiert das benutzerdefinierte Dialogfeld ein R-Script und führt es aus, um ein lineares Regressionsmodell zu erstellen und zu scoren. Hierzu wird die R-Funktion `lm` mit der hier dargestellten Signatur aufgerufen.

```
lm(formul, daten)
```

- *formul* gibt einen Ausdruck, wie beispielsweise `Na~Age` an. Dabei steht `Na` für das Zielfeld des Modells. Das Eingabefeld des Modells ist `Age`.
- *daten* steht für einen Datenrahmen, der die Werte der Felder enthält, die in der Formel angegeben sind.

Betrachten Sie ein benutzerdefiniertes Dialogfeld für Knoten mit einem einzelnen Steuerelement vom Typ "Feldauswahl", mit dem Sie das Eingabefeld des linearen Modells auswählen können. Die Scriptvorlage zum Generieren und Ausführen des R-Scripts, das das Modell erstellt, wird auf der Registerkarte **Script** eingegeben und kann folgendermaßen dargestellt sein:

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

- `%input%` ist der Wert der Eigenschaft "ID" des Steuerelements "Feldauswahl". Bei der Ausführung wird er durch den aktuellen Wert der Eigenschaft **Script** des Steuerelements ersetzt.
- Wenn Sie für die Eigenschaft **Script** des Steuerelements `%DieserWert%` festlegen, geben Sie an, dass der aktuelle Wert der Eigenschaft bei der Ausführung dem Wert des Steuerelements entspricht, nämlich dem Feld, das über die Feldauswahl ausgewählt wird.

Angenommen der Benutzer des benutzerdefinierten Dialogfelds für Knoten wählt das Feld "Age" als Eingabefeld des Modells aus. Dann wird das folgende R-Script durch das Dialogfeld für Knoten generiert:

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

Die Scriptvorlage zum Generieren und Ausführen des R-Scripts, das das Modell scort, wird auf der Registerkarte **Score-Script** eingegeben und kann folgendermaßen dargestellt sein:

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

Dieses R-Script hängt nicht von benutzerdefinierten Werten ab, sondern nur von dem Model, das mithilfe des R-Scripts für die Modellerstellung erstellt wurde. Daher wird das R-Script für die Modellerstellung so eingegeben, wie es im Feld **R-Modellscoringssyntax** des R-Erstellungsknotens eingegeben wird.

## Anzeigen der Vorschau eines benutzerdefinierten Dialogfelds für Knoten

Sie können eine Vorschau des Dialogfelds für Knoten anzeigen, das derzeit in der Dialogfelderstellung geöffnet sind. Die Anzeige und die Funktionen des Dialogfelds entsprechen denen bei der Ausführung aus einem Knoten in IBM SPSS Modeler.

- Feldauswahlen werden mit Pseudofeldern gefüllt.
- Über die Schaltfläche **OK** wird die Vorschau geschlossen.
- Wenn eine Hilfedatei angegeben ist, wird die Schaltfläche **Hilfe** aktiviert, über die die angegebene Datei geöffnet wird. Wenn keine Hilfedatei angegeben ist, ist die Schaltfläche "Hilfe" bei der Vorschau inaktiviert und bei der tatsächlichen Ausführung des Dialogfelds ausgeblendet.

Für die Vorschau eines benutzerdefinierten Dialogfelds für Knoten wählen Sie in den Menüs der Dialogfelderstellung **Datei > Dialogfeldvorschau** aus.

## Steuerelementtypen

Die Toolpalette stellt alle Standardsteuerelemente bereit, die in einem benutzerdefinierten Dialogfeld für Knoten erforderlich sein könnten.

- **Feldauswahl:** Eine Liste aller Felder aus dem aktiven Dataset. Weitere Informationen finden Sie in „Feldauswahl“.
- **Kontrollkästchen:** Ein einzelnes Kontrollkästchen. Weitere Informationen finden Sie in „Kontrollkästchen“ auf Seite 37.
- **Kombinationsfeld:** Ein Kombinationsfeld zur Erstellung von Dropdown-Listen. Weitere Informationen finden Sie in „Kombinationsfeld“ auf Seite 38.
- **Listenfeld:** Ein Listenfeld zur Erstellung von Einzelauswahl- oder Mehrfachauswahllisten. Weitere Informationen finden Sie in „Kombinationsfeld“ auf Seite 38.
- **Textsteuerelement:** Ein Textfeld, in das beliebiger Text eingegeben werden kann. Weitere Informationen finden Sie in „Textsteuerelement“ auf Seite 41.
- **Zahlensteuerelement:** Ein Textfeld, in das ausschließlich Zahlen eingegeben werden können. Weitere Informationen finden Sie in „Zahlensteuerelement“ auf Seite 42.
- **Steuerelement für Datum:** Ein Drehfeld für die Angabe von Datums-/Zeitwerten, beispielsweise Datum, Uhrzeit und Datum/Uhrzeit. Weitere Informationen finden Sie in „Steuerelement für Datum“ auf Seite 43.
- **Geschützter Text:** Ein Textfeld, das den Benutzereintrag durch Sterne maskiert. Weitere Informationen finden Sie in „Geschützter Text“ auf Seite 44.
- **Steuerelement für statischen Text:** Ein Steuerelement für die Anzeige von statischem Text. Weitere Informationen finden Sie in „Steuerelement für statischen Text“ auf Seite 45.
- **Farbauswahl:** Ein Steuerelement für die Angabe einer Farbe und für die Generierung des zugehörigen RGB-Werts. Weitere Informationen finden Sie in „Farbauswahl“ auf Seite 46.
- **Steuerelement für Tabelle:** Eine Tabelle mit einer festen Anzahl Spalten und einer variablen Anzahl Zeilen, die während der Ausführung eingefügt werden. Weitere Informationen finden Sie in „Steuerelement für Tabelle“ auf Seite 46.
- **Elementgruppe:** Ein Container für die Zusammenfassung von Steuerelementen wie beispielsweise Kontrollkästchen in Gruppen. Weitere Informationen finden Sie in „Elementgruppe“ auf Seite 49.
- **Optionsgruppe:** Eine Gruppe von Optionsfeldern. Weitere Informationen finden Sie in „Optionsgruppe“ auf Seite 49.
- **Kontrollkästchengruppe:** Ein Container für eine Gruppe von Steuerelementen, die über ein einziges Kontrollkästchen als Gruppe aktiviert oder inaktiviert werden. Weitere Informationen finden Sie in „Kontrollkästchengruppe“ auf Seite 51.
- **Dateisuche:** Ein Steuerelement, mit dem das Dateisystem durchsucht werden kann, um eine Datei zu öffnen oder zu speichern. Weitere Informationen finden Sie in „Dateisuche“ auf Seite 52.
- **Registerkarte:** Eine einzelne Registerkarte. Weitere Informationen finden Sie in „Tabulator“ auf Seite 53.
- **Unterdialogfeldschaltfläche:** Eine Schaltfläche, mit der ein Unterdialogfeld geöffnet wird. Weitere Informationen finden Sie in „Unterdialogfeldschaltfläche“ auf Seite 54.

## Feldauswahl

Das Steuerelement **Feldauswahl** zeigt die Liste der Felder an, die dem Endbenutzer des Dialogfelds für Knoten zur Verfügung stehen. Sie können alle Felder aus dem aktiven Dataset anzeigen (Standardeinstellung) oder die Liste nach Typ und Messniveau (zum Beispiel numerische Felder, die ein metrisches Messniveau aufweisen) filtern. Sie können auch eine andere Feldauswahl als Quelle der Felder für die aktuelle Feldauswahl angeben. Das Steuerelement "Feldauswahl" besitzt folgende Eigenschaften:

**ID.** Die eindeutige ID für das Steuerelement.

**Titel.** Ein optionaler Titel, der über dem Steuerelement angezeigt wird. Verwenden Sie bei mehrzeiligen Titeln \n, um Zeilenumbrüche anzugeben.

**Titelposition.** Legt die Position des Titels relativ zum Steuerelement fest. Es gibt die Werte **Oben** und **Links**. Die Standardeinstellung ist **Oben**. Diese Eigenschaft wird nur angewendet, wenn der Auswahltyp auf die Auswahl eines einzelnen Felds gesetzt ist.

**QuickInfo.** Optionaler QuickInfo-Text, der angezeigt wird, wenn der Benutzer die Maus über das Steuerelement bewegt. Der angegebene Text wird nur angezeigt, wenn Sie die Maus über den Titelbereich des Steuerelements bewegen. Wenn Sie die Maus über eines der aufgeführten Felder bewegen, werden der Name und die Beschriftung des Felds angezeigt.

**Mnemonisches Zeichen.** Ein optionales Zeichen im Titel, das als Tastenkombination für das Steuerelement verwendet werden kann. Das Zeichen wird im Titel unterstrichen. Die Tastenkombination wird aktiviert, indem Sie Alt + [mnemonisches Zeichen] drücken.

**Auswahltyp.** Gibt an, ob die Feldauswahl im benutzerdefinierten Dialogfeld für Knoten zur Auswahl eines einzelnen Felds oder mehrerer Felder aus der Feldliste verwendet werden kann.

**Trennzeichentyp.** Gibt das Trennzeichen zwischen den ausgewählten Feldern im generierten Script an. Die zulässigen Trennzeichen sind ein Leerzeichen, ein Komma und ein Pluszeichen (+). Sie können auch ein beliebiges einzelnes Zeichen für die Verwendung als Trennzeichen eingeben.

**Minimale Feldzahl.** Die minimale Anzahl Felder, die für ein Steuerelement, falls vorhanden, angegeben sein muss.

**Maximale Feldzahl.** Die maximale Anzahl Felder, die für ein Steuerelement, falls vorhanden, angegeben sein kann.

**Erforderlich für Ausführung.** Legt fest, ob ein Wert in diesem Steuerelement für die Fortsetzung der Ausführung erforderlich ist. Wenn **True** angegeben ist, muss der Benutzer des Dialogfelds für Knoten einen Wert für das Steuerelement angeben, da sonst das Anklicken der Schaltfläche **OK** zu einem Fehler führt. Wenn **False** angegeben ist, hat das Fehlen eines Werts für dieses Steuerelement keinerlei Auswirkungen auf die Schaltfläche **OK**.

**Variablenfilter.** Ermöglicht die Filterung des Sets von Feldern, die im Steuerelement angezeigt werden. Sie können nach Feldtyp und Messniveau filtern und festlegen, dass Mehrfachantwortsets in die Feldliste aufgenommen werden. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld "Filter" zu öffnen. Sie können das Dialogfeld **Filter** auch öffnen, indem Sie auf das Steuerelement **Feldauswahl** im Erstellungsbereich doppelklicken. Weitere Informationen finden Sie in „Filtern von Feldlisten“ auf Seite 37.

**Feldquelle.** Gibt an, dass eine andere Feldauswahl als Quelle der Felder für die aktuelle Feldauswahl dient. Wenn die Eigenschaft **Feldquelle** nicht festgelegt ist, dient das aktive Dataset als Quelle der Felder. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Feldquelle anzugeben.

**Script.** Gibt das Script an, das zur Laufzeit von diesem Steuerelement generiert und ausgeführt wird und in die Scriptvorlage eingefügt werden kann.

- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben. Klicken Sie bei mehrzeiligen Scripts oder langen Scripts auf die Schaltfläche mit Auslassungspunkten (...) und geben Sie Ihr Script in das Dialogfeld **Scripteigenschaft** ein.
- Der Wert `%DieserWert%` legt den Laufzeitwert des Steuerelements fest. Dabei handelt es sich um die Liste der Felder. Dies ist die Standardeinstellung.



**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

**Angeben der Feldquelle für eine Feldauswahl:** Im Dialogfeld **Feldquelle** wird die Quelle der Felder angegeben, die in der Feldauswahl angezeigt werden. Die Quelle kann eine beliebige andere Feldauswahl sein. Sie können auswählen, dass die im ausgewählten Steuerelement enthaltenen Felder oder die Felder aus dem aktiven Dataset angezeigt werden sollen, die nicht im ausgewählten Steuerelement enthalten sind.

## Filtern von Feldlisten

Über das Dialogfeld **Filter** für Steuerelemente für die Feldauswahl können Sie die Feldtypen aus dem aktiven Dataset filtern, die in den Listen angezeigt werden können. Sie können festlegen, ob Mehrfachantwortsets, die mit dem aktiven Dataset verknüpft sind, aufgenommen werden. Numerische Felder sind sämtliche numerischen Formate außer Datums- und Zeitformate.

## Kontrollkästchen

Das Steuerelement "Kontrollkästchen" ist ein einfaches Kontrollkästchen, das unterschiedliche R-Skripts oder Python for Spark-Skripts für den aktivierten und inaktivierten Status generieren und ausführen kann. Das Steuerelement "Kontrollkästchen" besitzt folgende Eigenschaften:

**ID.** Die eindeutige ID für das Steuerelement.

**Titel.** Ein optionaler Titel, der über dem Steuerelement angezeigt wird. Verwenden Sie bei mehrzeiligen Titeln \n, um Zeilenumbrüche anzugeben.

**QuickInfo.** Optionaler QuickInfo-Text, der angezeigt wird, wenn der Benutzer die Maus über das Steuerelement bewegt.

**Mnemonisches Zeichen.** Ein optionales Zeichen im Titel, das als Tastenkombination für das Steuerelement verwendet werden kann. Das Zeichen wird im Titel unterstrichen. Die Tastenkombination wird aktiviert, indem Sie Alt + [mnemonisches Zeichen] drücken.

**Standardwert.** Der standardmäßige Status des Kontrollkästchens, aktiviert oder inaktiviert.

**Script für Aktivierung/Inaktivierung.** Gibt das R-Script oder Python for Spark-Script an, das generiert und ausgeführt wird, wenn das Steuerelement aktiviert bzw. inaktiviert ist. Verwenden Sie den Wert der Eigenschaft **ID**, um das Script in die Scriptvorlage einzuschließen. Das generierte Script (aus der Eigenschaft **Script für Aktivierung** bzw. **Script für Inaktivierung**) wird an den angegebenen Positionen der ID eingefügt. Wenn z. B. die ID *Kontrollkästchen1* ist, werden während der Ausführung alle Instanzen von %%Kontrollkästchen1%% in der Scriptvorlage durch den Wert der Eigenschaft **Script für Aktivierung** ersetzt, wenn das Kontrollkästchen aktiviert ist, und durch den Wert der Eigenschaft **Script für Inaktivierung**, wenn das Kontrollkästchen inaktiviert ist.

- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben. Klicken Sie bei mehrzeiligen Scripts oder langen Scripts auf die Schaltfläche mit Auslassungspunkten (...) und geben Sie Ihr Script in das Dialogfeld **Scripteigenschaft** ein.

**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

## Kombinationsfeld

Mit dem Steuerelement "Kombinationsfeld" können Sie eine Dropdown-Liste erstellen, die ein R-Script oder Python for Spark-Script für den ausgewählten Listeneintrag generieren und ausführen kann. Sie ist auf eine Einzelauswahl beschränkt. Das Steuerelement "Kombinationsfeld" besitzt folgende Eigenschaften:

**ID.** Die eindeutige ID für das Steuerelement. Dies ist die ID, die für den Verweis auf das Steuerelement in der Scriptvorlage verwendet wird.

**Titel.** Ein optionaler Titel, der über dem Steuerelement angezeigt wird. Verwenden Sie bei mehrzeiligen Titeln \n, um Zeilenumbrüche anzugeben.

**Titelposition.** Legt die Position des Titels relativ zum Steuerelement fest. Es gibt die Werte **Oben** und **Links**. Die Standardeinstellung ist **Oben**.

**QuickInfo.** Optionaler QuickInfo-Text, der angezeigt wird, wenn der Benutzer die Maus über das Steuerelement bewegt.

**Listenelemente.** Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld "Listenelementeigenschaften" zu öffnen, in dem Sie die Listenelemente für das Steuerelement festlegen können. Sie können das Dialogfeld **Listenelementeigenschaften** auch öffnen, indem Sie auf das Steuerelement "Kombinationsfeld" im Erstellungsbereich doppelklicken.

**Mnemonicisches Zeichen.** Ein optionales Zeichen im Titel, das als Tastenkombination für das Steuerelement verwendet werden kann. Das Zeichen wird im Titel unterstrichen. Die Tastenkombination wird aktiviert, indem Sie Alt + [mnemonicisches Zeichen] drücken.

**Bearbeitbar.** Gibt an, ob das Steuerelement "Kombinationsfeld" bearbeitbar ist. Wenn das Steuerelement bearbeitbar ist, kann während der Ausführung ein benutzerdefinierter Wert eingegeben werden.

**Script.** Gibt das R- oder Python for Spark-Script an, das während der Ausführung von diesem Steuerelement generiert wird, und kann in die Scriptvorlage eingefügt werden.

- Der Wert %%DieserWert%% legt den Laufzeitwert des Steuerelements fest. Dabei handelt es sich um den Standardwert. Wenn die Listenelemente manuell definiert werden, ist der Laufzeitwert der Wert der Eigenschaft **Script** für das ausgewählte Listenelement. Wenn die Listenelemente auf einem Steuerelement für Ziellisten basieren, ist der Laufzeitwert der Wert des ausgewählten Listenelements. Bei Steuerelementen für Listenfelder mit Mehrfachauswahl ist der Laufzeitwert eine durch Leerzeichen getrennte Liste der ausgewählten Elemente. Weitere Informationen finden Sie in „Angaben von Listenelementen für Kombinationsfelder und Listenfelder“ auf Seite 39.
- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben. Klicken Sie bei mehrzeiligen Scripts oder langen Scripts auf die Schaltfläche mit Auslassungspunkten (...) und geben Sie Ihr Script in das Dialogfeld **Scripteigenschaft** ein.

**Handhabung von Anführungszeichen.** Gibt die Handhabung von Anführungszeichen im Laufzeitwert von %%DieserWert%% an, wenn die Scripteigenschaft %%DieserWert%% als Teil einer Zeichenfolge in Anführungszeichen enthält. In diesem Kontext ist eine Zeichenfolge in Anführungszeichen eine Zeichenfolge, die in einfache oder doppelte Anführungszeichen eingeschlossen ist. Die Handhabung von Anführungszeichen gilt nur für Anführungszeichen desselben Typs wie die Anführungszeichen, die %%DieserWert%% einschließen. Die folgenden Typen der Handhabung von Anführungszeichen sind verfügbar.

### Python

Enthält der Laufzeitwert von %%DieserWert%% Anführungszeichen, die auch als einschließende Anführungszeichen verwendet werden, werden die Anführungszeichen im Laufzeitwert durch einen umgekehrten Schrägstrich (\) entwertet. Wenn die Scripteigenschaft beispielsweise '%%DieserWert%%' ist und der Laufzeitwert des Kombinationsfelds

Combo box's value ist, lautet das generierte Script 'Combo box\'s value'. Beachten Sie, dass die Handhabung von Anführungszeichen nicht wirksam wird, wenn %%DieserWert%% in dreifache Anführungszeichen eingeschlossen ist.

- R** Enthält der Laufzeitwert von %%DieserWert%% Anführungszeichen, die auch als einschließende Anführungszeichen verwendet werden, werden die Anführungszeichen im Laufzeitwert durch einen umgekehrten Schrägstrich (\) entwertet. Wenn die Scripteigenschaft beispielsweise '%%DieserWert%%' ist und der Laufzeitwert des Kombinationsfelds Combo box's value ist, lautet das generierte Script 'Combo box\'s value'.
- Keine** Enthält der Laufzeitwert von %%DieserWert%% Anführungszeichen, die auch als einschließende Anführungszeichen verwendet werden, werden die Anführungszeichen unverändert beibehalten.

**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

**Angeben von Listenelementen für Kombinationsfelder und Listenfelder:** Im Dialogfeld "Listenelementeigenschaften" können Sie die Listenelemente für das Steuerelement für Kombinations- oder Listenfelder festlegen.

**Manuell definierte Werte** Ermöglicht das explizite Angeben der einzelnen Listenelemente.

- **ID.** Eine eindeutige ID für das Listenelement.
- **Name.** Der Name, der in der Liste für dieses Element angezeigt wird. Das Feld "Name" ist ein Pflichtfeld.
- **Standard.** Legt für ein Kombinationsfeld fest, ob das Listenelement das im Kombinationsfeld angezeigte Standardelement ist. Legt für ein Listenfeld fest, ob das Listenelement standardmäßig ausgewählt ist.
- **Script.** Gibt das R-Script oder das Python for Spark-Script an, das generiert wird, wenn das Listenelement ausgewählt wird.
- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben. Klicken Sie bei mehrzeiligen Scripts oder langen Scripts auf die Schaltfläche mit Auslassungspunkten (...) und geben Sie Ihr Script in das Dialogfeld **Scripteigenschaft** ein.

**Anmerkung:** Sie können ein neues Listenelement in die leere Zeile unten in der bestehenden Liste einfügen. Wenn Sie eine andere Eigenschaft als die ID eingeben, wird eine eindeutige ID generiert, die Sie übernehmen oder bearbeiten können. Sie können Listenelemente löschen, indem Sie auf die Zelle *ID* des Elements klicken und die Taste "Entf" drücken.

## Listenfeld

Das Steuerelement "Listenfeld" ermöglicht Ihnen die Anzeige einer Liste mit Elementen, die eine Einzel- oder Mehrfachauswahl unterstützen, und das Generieren eines R-Scripts oder Python for Spark-Scripts speziell für die ausgewählten Elemente. Das Steuerelement "Listenfeld" besitzt folgende Eigenschaften:

**ID.** Die eindeutige ID für das Steuerelement. Dies ist die ID, die für den Verweis auf das Steuerelement in der Scriptvorlage verwendet wird.

**Titel.** Ein optionaler Titel, der über dem Steuerelement angezeigt wird. Verwenden Sie bei mehrzeiligen Titeln \n, um Zeilenumbrüche anzugeben.

**QuickInfo.** Optionaler QuickInfo-Text, der angezeigt wird, wenn der Benutzer die Maus über das Steuerelement bewegt.

**Listenelemente.** Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld "Listenelementeigenschaften" zu öffnen, in dem Sie die Listenelemente für das Steuerelement festlegen können. Sie können das Dialogfeld **Listenelementeigenschaften** auch öffnen, indem Sie auf das Steuerelement "Listenfeld" im Erstellungsbereich doppelklicken.

**Mnemonicisches Zeichen.** Ein optionales Zeichen im Titel, das als Tastenkombination für das Steuerelement verwendet werden kann. Das Zeichen wird im Titel unterstrichen. Die Tastenkombination wird aktiviert, indem Sie Alt + [mnemonicisches Zeichen] drücken.

**Listenfeldtyp.** Legt fest, ob das Listenfeld ausschließlich eine Einzelauswahl oder die Mehrfachauswahl unterstützt. Sie können auch festlegen, dass Elemente als eine Liste von Kontrollkästchen angezeigt werden.

**Trennzeichentyp.** Gibt das Trennzeichen zwischen den ausgewählten Listenelementen im generierten Script an. Die zulässigen Trennzeichen sind ein Leerzeichen, ein Komma und ein Pluszeichen (+). Sie können auch ein beliebiges einzelnes Zeichen für die Verwendung als Trennzeichen eingeben.

**Minimale Auswahl.** Die minimale Anzahl Elemente, die im Steuerelement, falls vorhanden, ausgewählt sein muss.

**Maximale Auswahl.** Die maximale Anzahl Elemente, die im Steuerelement, falls vorhanden, ausgewählt sein kann.

**Script.** Gibt das R- oder Python for Spark-Script an, das während der Ausführung von diesem Steuerelement generiert wird, und kann in die Scriptvorlage eingefügt werden.

- Der Wert `%%DieserWert%` legt den Laufzeitwert des Steuerelements fest. Dabei handelt es sich um den Standardwert. Wenn die Listenelemente manuell definiert werden, ist der Laufzeitwert der Wert der Eigenschaft **Script** für das ausgewählte Listenelement. Wenn die Listenelemente auf einem Steuerelement für Ziellisten basieren, ist der Laufzeitwert der Wert des ausgewählten Listenelements. Bei Steuerelementen für Listenfelder mit Mehrfachauswahl ist der Laufzeitwert eine durch den angegebenen Trennzeichentyp getrennte Liste der ausgewählten Elemente (der Standardwert ist eine durch Leerzeichen getrennte Liste). Weitere Informationen finden Sie in „Angaben von Listenelementen für Kombinationsfelder und Listenfelder“ auf Seite 39.
- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben. Klicken Sie bei mehrzeiligen Scripts oder langen Scripts auf die Schaltfläche mit Auslassungspunkten (...) und geben Sie Ihr Script in das Dialogfeld **Scripteigenschaft** ein.

**Handhabung von Anführungszeichen.** Gibt die Handhabung von Anführungszeichen im Laufzeitwert von `%%DieserWert%` an, wenn die Scripteigenschaft `%%DieserWert%` als Teil einer Zeichenfolge in Anführungszeichen enthält. In diesem Kontext ist eine Zeichenfolge in Anführungszeichen eine Zeichenfolge, die in einfache oder doppelte Anführungszeichen eingeschlossen ist. Die Handhabung von Anführungszeichen gilt nur für Anführungszeichen desselben Typs wie die Anführungszeichen, die `%%DieserWert%` einschließen. Die folgenden Typen der Handhabung von Anführungszeichen sind verfügbar.

#### Python

Enthält der Laufzeitwert von `%%DieserWert%` Anführungszeichen, die auch als einschließende Anführungszeichen verwendet werden, werden die Anführungszeichen im Laufzeitwert durch einen umgekehrten Schrägstrich (\) entwertet. Wenn die Scripteigenschaft beispielsweise `'%%DieserWert%'` ist und das ausgewählte Listenelement `List item's value` ist, lautet das generierte Script `'List item\'s value'`. Beachten Sie, dass die Handhabung von Anführungszeichen nicht wirksam wird, wenn `%%DieserWert%` in dreifache Anführungszeichen eingeschlossen ist.

#### R

Enthält der Laufzeitwert von `%%DieserWert%` Anführungszeichen, die auch als einschließende Anführungszeichen verwendet werden, werden die Anführungszeichen im

Laufzeitwert durch einen umgekehrten Schrägstrich (\) entwertet. Wenn die Scripteigenschaft beispielsweise '%DieserWert%' ist und das ausgewählte Listenelement List item's value ist, lautet das generierte Script 'List item\'s value'.

**Keine** Enthält der Laufzeitwert von '%DieserWert%' Anführungszeichen, die auch als einschließende Anführungszeichen verwendet werden, werden die Anführungszeichen unverändert beibehalten.

**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

## Textsteuerelement

Das Textsteuerelement ist ein einfaches Textfeld, in das beliebiger Text eingegeben werden kann und das folgende Eigenschaften besitzt:

**ID.** Die eindeutige ID für das Steuerelement. Dies ist die ID, die für den Verweis auf das Steuerelement in der Scriptvorlage verwendet wird.

**Titel.** Ein optionaler Titel, der über dem Steuerelement angezeigt wird. Verwenden Sie bei mehrzeiligen Titeln \n, um Zeilenumbrüche anzugeben.

**Titelposition.** Legt die Position des Titels relativ zum Steuerelement fest. Es gibt die Werte **Oben** und **Links**. Die Standardeinstellung ist **Oben**.

**QuickInfo.** Optionaler QuickInfo-Text, der angezeigt wird, wenn der Benutzer die Maus über das Steuerelement bewegt.

**Mnemonisches Zeichen.** Ein optionales Zeichen im Titel, das als Tastenkombination für das Steuerelement verwendet werden kann. Das Zeichen wird im Titel unterstrichen. Die Tastenkombination wird aktiviert, indem Sie Alt + [mnemonisches Zeichen] drücken.

**Textinhalt.** Gibt an, ob ein beliebiger Text eingegeben werden kann oder das Textfeld eine Zeichenfolge enthalten muss, die die Regeln für IBM SPSS Modeler-Feldnamen beachtet.

**Standardwert.** Der Standardinhalt des Textfelds.

**Breite.** Gibt die Breite des Textbereichs des Steuerelements in Zeichen an. Als Werte sind positive Ganzzahlen zulässig. Ein leerer Wert bedeutet, dass die Breite automatisch festgelegt wird.

**Erforderlich für Ausführung.** Legt fest, ob ein Wert in diesem Steuerelement für die Fortsetzung der Ausführung erforderlich ist. Wenn **True** angegeben ist, muss der Benutzer des Dialogfelds für Knoten einen Wert für das Steuerelement angeben, da sonst das Anklicken der Schaltfläche **OK** zu einem Fehler führt. Wenn **False** angegeben ist, hat das Fehlen eines Werts für dieses Steuerelement keinerlei Auswirkungen auf die Schaltfläche **OK**. Standardmäßig ist dieser Wert **False**.

**Script.** Gibt das R-Script oder Python for Spark-Script an, das während der Ausführung von diesem Steuerelement generiert und ausgeführt wird und in die Scriptvorlage eingefügt werden kann.

- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben. Klicken Sie bei mehrzeiligen Scripts oder langen Scripts auf die Schaltfläche mit Auslassungspunkten (...) und geben Sie Ihr Script in das Dialogfeld **Scripteigenschaft** ein.
- Der Wert `%DieserWert%` legt den Laufzeitwert des Steuerelements fest. Dabei handelt es sich um den Inhalt des Textfelds. Dies ist die Standardeinstellung.
- Wenn die Scripteigenschaft `%DieserWert%` enthält und der Laufzeitwert des Textfelds leer ist, generiert das Textsteuerelement kein Script.

**Handhabung von Anführungszeichen.** Gibt die Handhabung von Anführungszeichen im Laufzeitwert von %%DieserWert%% an, wenn die Scripteigenschaft %%DieserWert%% als Teil einer Zeichenfolge in Anführungszeichen enthält. In diesem Kontext ist eine Zeichenfolge in Anführungszeichen eine Zeichenfolge, die in einfache oder doppelte Anführungszeichen eingeschlossen ist. Die Handhabung von Anführungszeichen gilt nur für Anführungszeichen desselben Typs wie die Anführungszeichen, die %%DieserWert%% einschließen. Die folgenden Typen der Handhabung von Anführungszeichen sind verfügbar.

**Python**

Enthält der Laufzeitwert von %%DieserWert%% Anführungszeichen, die auch als einschließende Anführungszeichen verwendet werden, werden die Anführungszeichen im Laufzeitwert durch einen umgekehrten Schrägstrich (\) entwertet. Wenn die Scripteigenschaft beispielsweise '%%DieserWert%%' ist und der Laufzeitwert des Textsteuerelements Text box's value ist, lautet das generierte Script 'Text box\'s value'. Die Handhabung von Anführungszeichen wird nicht wirksam, wenn %%DieserWert%% in dreifache Anführungszeichen eingeschlossen ist.

**R**

Enthält der Laufzeitwert von %%DieserWert%% Anführungszeichen, die auch als einschließende Anführungszeichen verwendet werden, werden die Anführungszeichen im Laufzeitwert durch einen umgekehrten Schrägstrich (\) entwertet. Wenn die Scripteigenschaft beispielsweise '%%DieserWert%%' ist und der Laufzeitwert des Textsteuerelements Text box's value ist, lautet das generierte Script 'Text box\'s value'.

**Keine**

Enthält der Laufzeitwert von %%DieserWert%% Anführungszeichen, die auch als einschließende Anführungszeichen verwendet werden, werden die Anführungszeichen unverändert beibehalten.

**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

## Zahlensteuerelement

Das Zahlensteuerelement ist ein Textfeld, in das ein numerischer Wert eingegeben werden kann und das folgende Eigenschaften besitzt:

**ID.** Die eindeutige ID für das Steuerelement. Dies ist die ID, die für den Verweis auf das Steuerelement in der Scriptvorlage verwendet wird.

**Titel.** Ein optionaler Titel, der über dem Steuerelement angezeigt wird. Verwenden Sie bei mehrzeiligen Titeln \n, um Zeilenumbrüche anzugeben.

**Titelposition.** Legt die Position des Titels relativ zum Steuerelement fest. Es gibt die Werte **Oben** und **Links**. Die Standardeinstellung ist **Oben**.

**QuickInfo.** Optionaler QuickInfo-Text, der angezeigt wird, wenn der Benutzer die Maus über das Steuerelement bewegt.

**Mnemonicisches Zeichen.** Ein optionales Zeichen im Titel, das als Tastenkombination für das Steuerelement verwendet werden kann. Das Zeichen wird im Titel unterstrichen. Die Tastenkombination wird aktiviert, indem Sie Alt + [mnemonicisches Zeichen] drücken.

**Numerischer Typ.** Legt etwaige Einschränkungen der Eingabemöglichkeiten fest. Der Wert "Reell" legt fest, dass es keine Einschränkungen für die Eingabe gibt, außer dass es sich um einen numerischen Wert handeln muss. Der Wert "Ganzzahl" legt fest, dass der Wert eine Ganzzahl sein muss.

**Drehfeldeingabe.** Gibt an, ob das Steuerelement als Drehfeld angezeigt wird. Der Standardwert ist **False**.

**Inkrement.** Das Inkrement, wenn das Steuerelement als Drehfeld angezeigt wird.

**Standardwert.** Der Standardwert (falls vorhanden).

**Minimum.** Der Mindestwert (falls vorhanden).

**Maximum.** Der Höchstwert (falls vorhanden).

**Breite.** Gibt die Breite des Textbereichs des Steuerelements in Zeichen an. Als Werte sind positive Ganzzahlen zulässig. Ein leerer Wert bedeutet, dass die Breite automatisch festgelegt wird.

**Erforderlich für Ausführung.** Legt fest, ob ein Wert in diesem Steuerelement für die Fortsetzung der Ausführung erforderlich ist. Wenn **True** angegeben ist, muss der Benutzer des Dialogfelds für Knoten einen Wert für das Steuerelement angeben, da sonst das Anklicken der Schaltfläche **OK** zu einem Fehler führt. Wenn **False** angegeben ist, hat das Fehlen eines Werts für dieses Steuerelement keinerlei Auswirkungen auf die Schaltfläche **OK**. Standardmäßig ist dieser Wert **False**.

**Script.** Gibt das R-Script oder Python for Spark-Script an, das während der Ausführung von diesem Steuerelement generiert und ausgeführt wird und in die Scriptvorlage eingefügt werden kann.

- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben. Klicken Sie bei mehrzeiligen Scripts oder langen Scripts auf die Schaltfläche mit Auslassungspunkten (...) und geben Sie Ihr Script in das Dialogfeld **Scripteigenschaft** ein.
- Der Wert `%%DieserWert%%` gibt den Laufzeitwert des Steuerelements an. Dabei handelt es sich um den numerischen Wert. Dies ist die Standardeinstellung.
- Wenn die R-Script-Eigenschaft `%%DieserWert%%` enthält und der Laufzeitwert des Zahlensteuerelements leer ist, generiert das Zahlensteuerelement kein Script.

**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

## Steuerelement für Datum

Das Steuerelement für Datum ist ein Drehfeld für die Angabe von Datums-/Zeitwerten, beispielsweise Datum, Uhrzeit und Datum/Uhrzeit. Das Steuerelement für Datum besitzt die folgenden Eigenschaften:

**ID.** Die eindeutige ID für das Steuerelement. Dies ist die ID, die für den Verweis auf das Steuerelement in der Scriptvorlage verwendet wird.

**Titel.** Ein optionaler Titel, der über dem Steuerelement angezeigt wird. Verwenden Sie bei mehrzeiligen Titeln `\n`, um Zeilenumbrüche anzugeben.

**Titelposition.** Legt die Position des Titels relativ zum Steuerelement fest. Es gibt die Werte **Oben** und **Links**. Die Standardeinstellung ist **Oben**.

**QuickInfo.** Optionaler QuickInfo-Text, der angezeigt wird, wenn der Benutzer die Maus über das Steuerelement bewegt.

**Mnemonisches Zeichen.** Ein optionales Zeichen im Titel, das als Tastenkombination für das Steuerelement verwendet werden kann. Das Zeichen wird im Titel unterstrichen. Die Tastenkombination wird aktiviert, indem Sie `Alt + [mnemonisches Zeichen]` drücken.

**Typ.** Gibt an, ob das Steuerelement für Datums-, Uhrzeit- oder Datums-/Zeitangaben dient.

**Datum**

Das Steuerelement gibt ein Kalenderdatum im Format yyyy-mm-tt an. Der Standardlaufzeitwert wird durch die Eigenschaft **Standardwert** angegeben.

**Zeit** Das Steuerelement gibt die Uhrzeit im Format hh:mm:ss an. Der Standardlaufzeitwert ist die aktuelle Uhrzeit.

**Datum/Uhrzeit**

Das Steuerelement gibt ein Datum und eine Uhrzeit im Format yyyy-mm-tt hh:mm:ss an. Der Standardlaufzeitwert ist das aktuelle Datum und die aktuelle Uhrzeit.

**Standardwert.** Der Standardlaufzeitwert des Steuerelements für den Typ **Datum**. Sie können angeben, dass das aktuelle Datum oder ein bestimmtes Datum angezeigt werden soll.

**Script.** Gibt das R-Script oder Python for Spark-Script an, das während der Ausführung von diesem Steuerelement generiert und ausgeführt wird und in die Scriptvorlage eingefügt werden kann.

- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben. Klicken Sie bei mehrzeiligen Scripts oder langen Scripts auf die Schaltfläche mit Auslassungspunkten (...) und geben Sie Ihr Script in das Dialogfeld **Scripteigenschaft** ein.
- Der Wert %%DieserWert%% gibt den Laufzeitwert des Steuerelements an. Dies ist die Standardeinstellung.

**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

**Anmerkung:** Das Steuerelement "Datum" wird in den Releases von IBM SPSS Modeler vor Release 18 nicht unterstützt.

**Geschützter Text**

Das Steuerelement "Geschützter Text" ist ein Textfeld, das den Benutzereintrag durch Sterne maskiert.

**ID.** Die eindeutige ID für das Steuerelement. Dies ist die ID, die für den Verweis auf das Steuerelement in der Scriptvorlage verwendet wird.

**Titel.** Ein optionaler Titel, der über dem Steuerelement angezeigt wird. Verwenden Sie bei mehrzeiligen Titeln \n, um Zeilenumbrüche anzugeben.

**Titelposition.** Legt die Position des Titels relativ zum Steuerelement fest. Es gibt die Werte **Oben** und **Links**. Die Standardeinstellung ist **Oben**.

**QuickInfo.** Optionaler QuickInfo-Text, der angezeigt wird, wenn der Benutzer die Maus über das Steuerelement bewegt.

**Mnemonisches Zeichen.** Ein optionales Zeichen im Titel, das als Tastenkombination für das Steuerelement verwendet werden kann. Das Zeichen wird im Titel unterstrichen. Die Tastenkombination wird aktiviert, indem Sie Alt + [mnemonisches Zeichen] drücken.

**Breite.** Gibt die Breite des Textbereichs des Steuerelements in Zeichen an. Als Werte sind positive Ganzzahlen zulässig. Ein leerer Wert bedeutet, dass die Breite automatisch festgelegt wird.

**Erforderlich für Ausführung.** Legt fest, ob ein Wert in diesem Steuerelement für die Fortsetzung der Ausführung erforderlich ist. Wenn **True** angegeben ist, muss der Benutzer des Dialogfelds für Knoten einen Wert für das Steuerelement angeben, da sonst das Anklicken der Schaltfläche **OK** zu einem Fehler führt. Wenn **False** angegeben ist, hat das Fehlen eines Werts für dieses Steuerelement keinerlei Auswirkungen auf die Schaltfläche **OK**. Standardmäßig ist dieser Wert **False**.



**Script.** Gibt das R-Script oder Python for Spark-Script an, das während der Ausführung von diesem Steuerelement generiert und ausgeführt wird und in die Scriptvorlage eingefügt werden kann.

- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben. Klicken Sie bei mehrzeiligen Scripts oder langen Scripts auf die Schaltfläche mit Auslassungspunkten (...) und geben Sie Ihr Script in das Dialogfeld **Scripteigenschaft** ein.
- Der Wert `%%DieserWert%%` legt den Laufzeitwert des Steuerelements fest. Dabei handelt es sich um den Inhalt des Textfelds. Dies ist die Standardeinstellung.
- Wenn die Scripteigenschaft `%%DieserWert%%` enthält und der Laufzeitwert des Steuerelements **Geschützter Text** leer ist, generiert das Steuerelement "Geschützter Text" kein R-Script oder Python for Spark-Script.

**Handhabung von Anführungszeichen.** Gibt die Handhabung von Anführungszeichen im Laufzeitwert von `%%DieserWert%%` an, wenn die Scripteigenschaft `%%DieserWert%%` als Teil einer Zeichenfolge in Anführungszeichen enthält. In diesem Kontext ist eine Zeichenfolge in Anführungszeichen eine Zeichenfolge, die in einfache oder doppelte Anführungszeichen eingeschlossen ist. Die Handhabung von Anführungszeichen gilt nur für Anführungszeichen desselben Typs wie die Anführungszeichen, die `%%DieserWert%%` einschließen, und nur wenn Übergebenen Wert `verschlüsseln=False` gesetzt ist. Die folgenden Typen der Handhabung von Anführungszeichen sind verfügbar.

#### **Python**

Enthält der Laufzeitwert von `%%DieserWert%%` Anführungszeichen, die auch als einschließende Anführungszeichen verwendet werden, werden die Anführungszeichen im Laufzeitwert durch einen umgekehrten Schrägstrich (\) entwertet. Wenn die Scripteigenschaft beispielsweise `'%%DieserWert%%'` ist und der Laufzeitwert des Steuerelements `Secured Text's value` ist, lautet das generierte Script `'Secured Text\'s value'`. Die Handhabung von Anführungszeichen wird nicht wirksam, wenn `%%DieserWert%%` in dreifache Anführungszeichen eingeschlossen ist.

**R** Enthält der Laufzeitwert von `%%DieserWert%%` Anführungszeichen, die auch als einschließende Anführungszeichen verwendet werden, werden die Anführungszeichen im Laufzeitwert durch einen umgekehrten Schrägstrich (\) entwertet. Wenn die Scripteigenschaft beispielsweise `'%%DieserWert%%'` ist und der Laufzeitwert des Steuerelements `Secured Text's value` ist, lautet das generierte Script `'Secured Text\'s value'`.

**Keine** Enthält der Laufzeitwert von `%%DieserWert%%` Anführungszeichen, die auch als einschließende Anführungszeichen verwendet werden, werden die Anführungszeichen unverändert beibehalten.

**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

**Anmerkung:** Das Steuerelement "Geschützter Text" wird in den Releases von IBM SPSS Modeler vor Release 18 nicht unterstützt.

#### **Steuerelement für statischen Text**

Über das Steuerelement für statischen Text können Sie einen Textblock in Ihr Dialogfeld für Knoten einfügen. Es besitzt folgende Eigenschaften:

**ID.** Die eindeutige ID für das Steuerelement.

**Titel.** Der Inhalt des Textblocks. Verwenden Sie bei mehrzeiligen Inhalten `\n`, um Zeilenumbrüche zu markieren.

**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

## Farbauswahl

Das Steuerelement "Farbauswahl" ist eine Benutzerschnittstelle für die Angabe einer Farbe und für die Generierung des zugehörigen RGB-Werts. Das Steuerelement "Farbauswahl" besitzt die folgenden Eigenschaften:

**ID.** Die eindeutige ID für das Steuerelement. Dies ist die ID, die für den Verweis auf das Steuerelement in der Scriptvorlage verwendet wird.

**Titel.** Ein optionaler Titel, der über dem Steuerelement angezeigt wird. Verwenden Sie bei mehrzeiligen Titeln \n, um Zeilenumbrüche anzugeben.

**Titelposition.** Legt die Position des Titels relativ zum Steuerelement fest. Es gibt die Werte **Oben** und **Links**. Die Standardeinstellung ist **Oben**.

**QuickInfo.** Optionaler QuickInfo-Text, der angezeigt wird, wenn der Benutzer die Maus über das Steuerelement bewegt.

**Mnemonicisches Zeichen.** Ein optionales Zeichen im Titel, das als Tastenkombination für das Steuerelement verwendet werden kann. Das Zeichen wird im Titel unterstrichen. Die Tastenkombination wird aktiviert, indem Sie Alt + [mnemonicisches Zeichen] drücken.

**Script.** Gibt das R-Script oder Python for Spark-Script an, das während der Ausführung von diesem Steuerelement generiert und ausgeführt wird und in die Scriptvorlage eingefügt werden kann.

- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben. Klicken Sie bei mehrzeiligen Scripts oder langen Scripts auf die Schaltfläche mit Auslassungspunkten (...) und geben Sie Ihr Script in das Dialogfeld **Scripteigenschaft** ein.
- Der Wert `%DieserWert%` gibt den Laufzeitwert des Steuerelements an. Dabei handelt es sich um den RGB-Wert der ausgewählten Farbe. Der RGB-Wert wird als eine durch Leerzeichen getrennte Liste mit ganzen Zahlen in der folgenden Reihenfolge dargestellt: R-Wert, G-Wert, B-Wert.

**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

**Anmerkung:** Das Steuerelement "Farbauswahl" wird in den Releases von IBM SPSS Modeler vor Release 18 nicht unterstützt.

## Steuerelement für Tabelle

Das Tabellensteuerelement erstellt eine Tabelle mit einer festen Anzahl Spalten und einer variablen Anzahl Zeilen, die während der Ausführung eingefügt wurden. Das Tabellensteuerelement besitzt folgende Eigenschaften:

**ID.** Die eindeutige ID für das Steuerelement. Dies ist die ID, die für den Verweis auf das Steuerelement in der Scriptvorlage verwendet wird.

**Titel.** Ein optionaler Titel, der über dem Steuerelement angezeigt wird. Verwenden Sie bei mehrzeiligen Titeln \n, um Zeilenumbrüche anzugeben.

**QuickInfo.** Optionaler QuickInfo-Text, der angezeigt wird, wenn der Benutzer die Maus über das Steuerelement bewegt.

**Mnemonisches Zeichen.** Ein optionales Zeichen im Titel, das als Tastenkombination für das Steuerelement verwendet werden kann. Das Zeichen wird im Titel unterstrichen. Die Tastenkombination wird aktiviert, indem Sie Alt + [mnemonisches Zeichen] drücken.

**Umordnungsschaltflächen.** Gibt an, ob der Tabelle die Schaltflächen zum Verschieben nach oben und nach unten hinzugefügt werden. Mit diesen Schaltflächen kann die Reihenfolge der Tabellenzeilen während der Ausführung geändert werden.

**Tabellenspalten.** Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld **Tabellenspalten** zu öffnen, in dem Sie die Spalten der Tabelle angeben können.

**Minimale Zeilenzahl.** Die minimale Anzahl Zeilen, die die Tabelle enthalten muss.

**Maximale Zeilenzahl.** Die maximale Anzahl Zeilen, die die Tabelle enthalten kann.

**Erforderlich für Ausführung.** Legt fest, ob ein Wert in diesem Steuerelement für die Fortsetzung der Ausführung erforderlich ist. Wenn **True** angegeben ist, muss der Benutzer des Dialogfelds für Knoten einen Wert für das Steuerelement angeben, da sonst das Anklicken der Schaltfläche **OK** zu einem Fehler führt. Wenn **False** angegeben ist, hat das Fehlen eines Werts für dieses Steuerelement keinerlei Auswirkungen auf die Schaltfläche **OK**.

**Script.** Gibt das R- oder Python for Spark-Script an, das während der Ausführung von diesem Steuerelement generiert wird, und kann in die Scriptvorlage eingefügt werden.

- Der Wert `%%DieserWert%` legt den Laufzeitwert des Steuerelements fest. Dabei handelt es sich um den Standardwert. Der Laufzeitwert ist eine durch Leerzeichen getrennte Liste des Scripts, das von jeder Spalte in der Tabelle ab der Spalte links außen generiert wird. Wenn die Scripteigenschaft `%%DieserWert%` enthält und von keiner der Spalten ein Script generiert wird, generiert die Tabelle als Ganzes kein Script.
- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben. Klicken Sie bei mehrzeiligen Scripts oder langen Scripts auf die Schaltfläche mit Auslassungspunkten (...) und geben Sie Ihr Script in das Dialogfeld **Scripteigenschaft** ein.

**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

**Anmerkung:** Das Steuerelement "Tabelle" wird in den Releases von IBM SPSS Modeler vor Release 18 nicht unterstützt.

**Angaben von Spalten für Tabellensteuerelemente:** Das Dialogfeld **Tabellenspalten** gibt die Eigenschaften der Spalten des Tabellensteuerelements an.

**ID.** Eine eindeutige ID für die Spalte.

**Spaltenname.** Der Name der Spalte, wie er in der Tabelle angezeigt wird.

**Inhalt.** Gibt den Datentyp für die Spalte an. Der Wert **Reelle Zahl** gibt an, dass es keine Einschränkungen für den eingegebenen Wert gibt, außer dass es sich um einen numerischen Wert handeln muss. Der Wert **Ganze Zahl** gibt an, dass der Wert eine ganze Zahl sein muss. Der Wert **Beliebig** gibt an, dass es keine Einschränkungen für den eingegebenen Wert gibt. Der Wert **Variablenname** gibt an, dass der Wert die Anforderungen für einen gültigen Variablennamen in IBM SPSS Statistics erfüllen muss.

**Standardwert.** Der Standardwert für diese Spalte, falls vorhanden, wenn der Tabelle während der Ausführung neue Zeilen hinzugefügt werden.

**Trennzeichentyp.** Gibt das Trennzeichen zwischen den Werten der Spalte im generierten Script an. Die zulässigen Trennzeichen sind ein Leerzeichen, ein Komma und ein Pluszeichen (+). Sie können auch ein beliebiges einzelnes Zeichen für die Verwendung als Trennzeichen eingeben.

**In Anführungszeichen.** Gibt an, ob jeder Wert in der Spalte im generierten Script in Anführungszeichen eingeschlossen ist.

**Handhabung von Anführungszeichen.** Gibt die Handhabung von Anführungszeichen in den Zelleneinträgen für die Spalte an, wenn die Eigenschaft **In Anführungszeichen** wahr ist. Die Handhabung von Anführungszeichen gilt nur für Anführungszeichen in Zellenwerten. Die folgenden Typen der Handhabung von Anführungszeichen sind verfügbar.

**Python**

Anführungszeichen in Zellenwerten werden durch einen umgekehrten Schrägstrich (\) entwertet. Lautet der Zellenwert beispielsweise Dieser Wert in "Anführungszeichen", lautet das generierte Script "Dieser Wert in \"Anführungszeichen\"".

**R** Anführungszeichen in Zellenwerten werden durch einen umgekehrten Schrägstrich (\) entwertet. Lautet der Zellenwert beispielsweise Dieser Wert in "Anführungszeichen", lautet das generierte Script "Dieser Wert in \"Anführungszeichen\"".

**Keine** Anführungszeichen in Zellenwerten werden unverändert beibehalten.

**Breite (Zeichen).** Gibt die Breite der Spalte in Zeichen an. Als Werte sind positive Ganzzahlen zulässig.

**Script.** Gibt das R-Script oder Python for Spark-Script an, das von dieser Spalte während der Ausführung generiert wird. Das generierte Script für die ganze Tabelle ist eine durch Leerzeichen getrennte Liste des Scripts, das von jeder Spalte in der Tabelle ab der Spalte links außen generiert wird.

- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben. Klicken Sie bei mehrzeiligen Scripts oder langen Scripts auf die Schaltfläche mit Auslassungspunkten (...) und geben Sie Ihr Script in das Dialogfeld **Scripteigenschaft** ein.
- Der Wert %%DieserWert%% legt den Laufzeitwert der Spalte fest. Dabei handelt es sich um eine Liste der Werte in der Spalte, die durch das angegebene Trennzeichen getrennt werden.
- Wenn die Scripteigenschaft für die Spalte %%DieserWert%% enthält und der Laufzeitwert der Spalte leer ist, generiert die Spalte kein Script.

**Anmerkung:** Sie können eine Zeile für eine neue Tabellenspalte in die leere Zeile unten in der bestehenden Liste im Dialogfeld **Tabellenspalten** einfügen. Wenn Sie andere Eigenschaften als die ID eingeben, wird eine eindeutige ID generiert, die Sie übernehmen oder bearbeiten können. Sie können eine Tabellenspalte löschen, indem Sie auf die Zellen-ID für die Tabellenspalte klicken und die Löschtaste drücken.

**Link zu Steuerelement**

Sie können ein Tabellensteuerelement mit einem Steuerelement "Feldauswahl" verknüpfen. Wenn ein Tabellensteuerelement mit einer Feldauswahl verknüpft ist, gibt es für jedes Feld in der Feldauswahl eine Zeile in der Tabelle. Der Tabelle werden Zeilen hinzugefügt, indem der Feldauswahl Felder hinzugefügt werden. Zeilen werden aus der Tabelle gelöscht, indem Felder aus der Feldauswahl entfernt werden. Mit einem verknüpften Tabellensteuerelement können zum Beispiel Eigenschaften von Feldern angegeben werden, die in einer Feldauswahl ausgewählt sind.

Wenn Sie die Verknüpfung aktivieren wollen, muss die Tabelle eine Spalte enthalten, die als Inhaltseigenschaft einen Variablennamen aufweist und im Erstellungsbereich muss mindestens ein Steuerelement "Feldauswahl" vorhanden sein.

Wenn Sie ein Tabellensteuerelement mit einer Feldauswahl verknüpfen wollen, geben Sie die Feldauswahl aus der Liste der verfügbaren Steuerelemente in der Gruppe **Link zu Steuerelement** im Dialogfeld **Tabellenspalten** an. Wählen Sie dann die Tabellenspalte aus, die die Verknüpfung definiert. Diese Spalte wird auch als **verlinkte Spalte** bezeichnet. Beim Rendern der Tabelle zeigt die verlinkte Spalte die aktuellen Felder in der Feldauswahl an. Sie können nur Verknüpfungen zu Feldauswahlen mit mehreren Feldern erstellen.

## Elementgruppe

Das Steuerelement "Elementgruppe" ist ein Container für andere Steuerelemente, mit dem Sie das Script, das aus mehreren Steuerelementen generiert wird, in Gruppen zusammenfassen und steuern. Gehen wir beispielsweise von einer Gruppe von Kontrollkästchen aus, die optionale Einstellungen für einen Unterbefehl festlegen. Das Script für den Unterbefehl soll jedoch nur erstellt werden, wenn mindestens ein Kontrollkästchen aktiviert ist. Das erreichen Sie, indem Sie das Steuerelement "Elementgruppe" als Container für die Steuerelemente der Kontrollkästchen verwenden. Die folgenden Steuerelementtypen können in einer Elementgruppe enthalten sein: Feldauswahl, Kontrollkästchen, Kombinationsfeld, Listenfeld, Textsteuerelement, Zahlensteuerelement, Steuerelement für statischen Text, Optionsfeldgruppe und Dateibrowser. Das Steuerelement für Elementgruppen besitzt folgende Eigenschaften:

**ID.** Die eindeutige ID für das Steuerelement. Dies ist die ID, die für den Verweis auf das Steuerelement in der Scriptvorlage verwendet wird.

**Titel.** Ein optionaler Titel für die Gruppe. Verwenden Sie bei mehrzeiligen Titeln \n, um Zeilenumbrüche zu markieren.

**Script.** Gibt das R-Script oder Python for Spark-Script an, das während der Ausführung von diesem Steuerelement generiert und ausgeführt wird und in die Scriptvorlage eingefügt werden kann.

- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben. Klicken Sie bei mehrzeiligen Scripts oder langen Scripts auf die Schaltfläche mit Auslassungspunkten (...) und geben Sie Ihr Script in das Dialogfeld **Scripteigenschaft** ein.
- Sie können IDs für beliebige Steuerelemente aus der Elementgruppe auswählen. Während der Ausführung werden die IDs durch das R-Script oder Python-Script ersetzt, das von den Steuerelementen generiert wird.
- Der Wert `%%DieserWert%` generiert eine durch Leerzeichen getrennte Liste des R-Scripts oder Python-Scripts, das von den einzelnen Steuerelementen in der Elementgruppe generiert wird, und zwar in der Reihenfolge, in der sie in der Gruppe angezeigt werden (von oben nach unten). Dies ist die Standard-einstellung. Wenn die Scripteigenschaft `%%DieserWert%` enthält und von keinem der Steuerelemente in der Elementgruppe ein Script generiert wird, generiert die Elementgruppe als Ganzes kein Script.

**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

## Optionsgruppe

Das Steuerelement "Optionsgruppe" ist ein Container für eine Gruppe von Optionsfeldern, von denen jedes eine Gruppe von verschachtelten Steuerelementen enthalten kann. Das Steuerelement "Optionsgruppe" besitzt folgende Eigenschaften:

**ID.** Die eindeutige ID für das Steuerelement. Dies ist die ID, die für den Verweis auf das Steuerelement in der Scriptvorlage verwendet wird.

**Titel.** Ein optionaler Titel für die Gruppe. Verwenden Sie bei mehrzeiligen Titeln \n, um Zeilenumbrüche zu markieren.

**QuickInfo.** Optionaler QuickInfo-Text, der angezeigt wird, wenn der Benutzer die Maus über das Steuerelement bewegt.

**Optionsfelder.** Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld "Optionsgruppeneigenschaften" zu öffnen, in dem Sie die Eigenschaften der Optionsfelder festlegen und Felder aus der Gruppe entfernen bzw. in die Gruppe aufnehmen können. Die Möglichkeit, Steuerelemente unter einem bestimmten Optionsfeld zu verschachteln, ist eine Eigenschaft des Optionsfelds, die im Dialogfeld "Optionsgruppeneigenschaften" eingestellt wird. Beachten Sie, dass Sie das Dialogfeld **Optionsgruppeneigenschaften** auch öffnen können, indem Sie auf das Steuerelement "Optionsgruppe" im Erstellungsreich doppelklicken.

**Script.** Gibt das R-Script oder Python for Spark-Script an, das während der Ausführung von diesem Steuerelement generiert wird und in die Scriptvorlage eingefügt werden kann.

- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben. Klicken Sie bei mehrzeiligen Scripts oder langen Scripts auf die Schaltfläche mit Auslassungspunkten (...) und geben Sie Ihr Script in das Dialogfeld **Scripteigenschaft** ein.
- Der Wert `%%DieserWert%%` gibt den Laufzeitwert der Optionsfeldgruppe an. Dabei handelt es sich um den Wert der Scripteigenschaft für das gewählte Optionsfeld. Dies ist die Standardeinstellung. Wenn die Scripteigenschaft `%%DieserWert%%` enthält und vom ausgewählten Optionsfeld kein Script generiert wird, generiert die Optionsfeldgruppe kein Script.

**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

**Definieren von Optionsfeldern:** Über das Dialogfeld "Optionsfeldgruppen - Eigenschaften" können Sie eine Gruppe von Optionsfeldern festlegen.

**ID.** Eine eindeutige ID für das Optionsfeld.

**Spaltenname.** Der Name, der neben dem Optionsfeld angezeigt wird. Das Feld "Name" ist ein Pflichtfeld.

**QuickInfo.** Optionaler QuickInfo-Text, der angezeigt wird, wenn der Benutzer die Maus über das Steuerelement bewegt.

**Mnemonisches Zeichen.** Ein optionales Zeichen im Namen, das als mnemonisches Zeichen verwendet wird. Das angegebene Zeichen muss im Namen vorhanden sein.

**Verschachtelte Gruppe.** Gibt an, ob andere Steuerelemente unter diesem Optionsfeld verschachtelt werden können. Der Standardwert ist **False**. Wenn der Wert für die Eigenschaft "Verschachtelte Gruppe" auf "True" eingestellt ist, wird unter dem zugehörigen Optionsfeld ein rechteckiger Ablegebereich angezeigt, und zwar verschachtelt und eingerückt. Die folgenden Steuerelemente können unter einem Optionsfeld verschachtelt sein: Feldauswahl, Kontrollkästchen, Textsteuerelement, Steuerelement für statischen Text, Zahlensteuerelement, Kombinationsfeld, Listenfeld und Dateibrowser.

**Standard.** Legt fest, ob es sich beim Optionsfeld um die Standardauswahl handelt.

**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

**Script.** Gibt das R-Script oder das Python for Spark-Script an, das generiert wird, wenn das Optionsfeld ausgewählt wird.

- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben. Klicken Sie bei mehrzeiligen Scripts oder langen Scripts auf die Schaltfläche mit Auslassungspunkten (...) und geben Sie Ihr Script in das Dialogfeld **Scripteigenschaft** ein.
- Für Optionsfelder mit verschachtelten Steuerelementen generiert der Wert %%DieserWert%% eine durch Leerzeichen getrennte Liste mit dem von den einzelnen verschachtelten Steuerelementen generierten R-Script oder Python for Spark-Script und zwar in der Reihenfolge, in der sie unter dem Optionsfeld auftreten (von oben nach unten).

Sie können ein neues Optionsfeld in die leere Zeile unten in der bestehenden Liste einfügen. Wenn Sie eine andere Eigenschaft als die ID eingeben, wird eine eindeutige ID generiert, die Sie übernehmen oder bearbeiten können. Sie können Optionsfelder löschen, indem Sie auf die Zelle *ID* des Felds klicken und die Löschtaste drücken.

## Kontrollkästchengruppe

Das Steuerelement "Kontrollkästchengruppe" ist ein Container für eine Gruppe von Steuerelementen, die über ein einziges Kontrollkästchen als Gruppe aktiviert oder inaktiviert werden. Die folgenden Steuerelementtypen können in einer Kontrollkästchengruppe vorhanden sein: Feldauswahl, Kontrollkästchen, Kombinationsfeld, Listenfeld, Textsteuerelement, Zahlensteuerelement, Steuerelement für statischen Text, Optionsfeldgruppe und Dateibrowser. Das Steuerelement "Kontrollkästchengruppe" besitzt folgende Eigenschaften:

**ID.** Die eindeutige ID für das Steuerelement. Dies ist die ID, die für den Verweis auf das Steuerelement in der Scriptvorlage verwendet wird.

**Titel.** Ein optionaler Titel für die Gruppe. Verwenden Sie bei mehrzeiligen Titeln \n, um Zeilenumbrüche zu markieren.

**Kontrollkästchentitel.** Die optionale Beschriftung, die mit dem Kontrollkästchen angezeigt wird. Für Zeilenumbrüche kann \n verwendet werden.

**QuickInfo.** Optionaler QuickInfo-Text, der angezeigt wird, wenn der Benutzer die Maus über das Steuerelement bewegt.

**Mnemonisches Zeichen.** Ein optionales Zeichen im Titel, das als Tastenkombination für das Steuerelement verwendet werden kann. Das Zeichen wird im Titel unterstrichen. Die Tastenkombination wird aktiviert, indem Sie Alt + [mnemonisches Zeichen] drücken.

**Standardwert.** Der standardmäßige Status des Kontrollkästchens, aktiviert oder inaktiviert.

**R-Script für Aktivierung/Inaktivierung.** Gibt das R-Script an, das generiert wird, wenn das Steuerelement aktiviert bzw. inaktiviert ist. Verwenden Sie den Wert der Eigenschaft **ID**, um das R-Script in die Scriptvorlage einzuschließen. Das generierte R-Script (aus der Eigenschaft **R-Script für Aktivierung** bzw. **R-Script für Inaktivierung**) wird an den angegebenen Positionen der ID eingefügt. Wenn z. B. die ID *Kontrollkästchengruppe1* ist, werden während der Ausführung alle Instanzen von %%Kontrollkästchengruppe1%% in der Scriptvorlage durch den Wert der Eigenschaft **R-Script für Aktivierung** ersetzt, wenn das Kontrollkästchen aktiviert ist, und durch den Wert der Eigenschaft **R-Script für Inaktivierung**, wenn das Kontrollkästchen inaktiviert ist.

- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben. Klicken Sie bei mehrzeiligen Scripts oder langen Scripts auf die Schaltfläche mit Auslassungspunkten (...) und geben Sie Ihr Script in das Dialogfeld **Scripteigenschaft** ein.
- Sie können IDs für beliebige Steuerelemente aus der Kontrollkästchengruppe auswählen. Während der Ausführung werden die IDs durch das R-Script ersetzt, das von den Steuerelementen generiert wird.
- Der Wert %%DieserWert%% kann in der Eigenschaft **R-Script für Aktivierung** oder in der Eigenschaft **R-Script für Inaktivierung** verwendet werden. Er generiert eine durch Leerzeichen getrennte Liste des R-

Scripts, das von den einzelnen Steuerelementen in der Kontrollkästchengruppe generiert wird, und zwar in der Reihenfolge, in der sie in der Gruppe angezeigt werden (von oben nach unten).

- Standardmäßig besitzt die Eigenschaft **R-Script für Aktivierung** den Wert von %%DieserWert%% und die Eigenschaft **R-Script für Inaktivierung** ist leer.

**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

## Dateisuche

Das Steuerelement "Dateisuche" setzt sich aus einem Textfeld für einen Dateipfad und der Schaltfläche "Durchsuchen" zusammen, über die ein IBM SPSS Modeler-Standarddialogfeld zum Öffnen und Speichern von Dateien geöffnet wird. Das Steuerelement "Dateisuche" besitzt folgende Eigenschaften:

**ID.** Die eindeutige ID für das Steuerelement. Dies ist die ID, die für den Verweis auf das Steuerelement in der Scriptvorlage verwendet wird.

**Titel.** Ein optionaler Titel, der über dem Steuerelement angezeigt wird. Verwenden Sie bei mehrzeiligen Titeln \n, um Zeilenumbrüche anzugeben.

**Titelposition.** Legt die Position des Titels relativ zum Steuerelement fest. Es gibt die Werte **Oben** und **Links**. Die Standardeinstellung ist **Oben**.

**QuickInfo.** Optionaler QuickInfo-Text, der angezeigt wird, wenn der Benutzer die Maus über das Steuerelement bewegt.

**Mnemonicisches Zeichen.** Ein optionales Zeichen im Titel, das als Tastenkombination für das Steuerelement verwendet werden kann. Das Zeichen wird im Titel unterstrichen. Die Tastenkombination wird aktiviert, indem Sie Alt + [mnemonicisches Zeichen] drücken.

**Dateisystemoperation.** Legt fest, ob das über die Schaltfläche "Durchsuchen" geöffnete Dialogfeld zum Öffnen oder Speichern von Dateien geeignet ist. Der Wert "Öffnen" gibt an, dass über das Dialogfeld die Existenz einer angegebenen Datei überprüft wird. Der Wert "Speichern" gibt an, dass über das Dialogfeld die Existenz einer angegebenen Datei nicht überprüft wird.

**Suchtyp.** Legt fest, ob das Dialogfeld verwendet wird, um eine Datei (Datei suchen) oder einen Ordner (Ordner suchen) auszuwählen.

**Dateifilter.** Klicken Sie auf die Schaltfläche mit den Auslassungspunkten (...), um das Dialogfeld "Dateifilter" zu öffnen, in dem Sie die verfügbaren Dateitypen für das Dialogfeld "Öffnen" oder "Speichern" festlegen können. Standardmäßig sind alle Dateitypen verfügbar. Beachten Sie, dass Sie das Dialogfeld **Dateifilter** auch öffnen können, indem Sie auf das Steuerelement "Dateisuche" im Erstellungsbereich doppelklicken.

**Dateisystemtyp.** Im Modus für verteilte Analysen gibt der Dateisystemtyp an, ob das Dialogfeld "Öffnen" oder "Speichern" das Dateisystem durchsucht, auf dem der IBM SPSS Modeler-Server ausgeführt wird, oder das Dateisystem Ihres lokalen Computers. Wählen Sie **Server**, um das Dateisystem des Servers zu durchsuchen, oder **Client**, um das Dateisystem Ihres lokalen Computers zu durchsuchen. Im Modus für lokale Analysen hat diese Eigenschaft keinerlei Auswirkungen.

**Erforderlich für Ausführung.** Legt fest, ob ein Wert in diesem Steuerelement für die Fortsetzung der Ausführung erforderlich ist. Wenn **True** angegeben ist, muss der Benutzer des Dialogfelds für Knoten einen Wert für das Steuerelement angeben, da sonst das Anklicken der Schaltfläche **OK** zu einem Fehler führt. Wenn **False** angegeben ist, hat das Fehlen eines Werts für dieses Steuerelement keinerlei Auswirkungen auf die Schaltfläche **OK**. Standardmäßig ist dieser Wert **False**.



**Standard.** Der Standardwert des Steuerelements.

**Script.** Gibt das R-Script oder Python for Spark-Script an, das während der Ausführung von diesem Steuerelement generiert wird und in die Scriptvorlage eingefügt werden kann.

- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben. Klicken Sie bei mehrzeiligen Scripts oder langen Scripts auf die Schaltfläche mit Auslassungspunkten (...) und geben Sie Ihr Script in das Dialogfeld **Scripteigenschaft** ein.
- Der Wert `%%DieserWert%%` legt den Laufzeitwert des Textfelds fest, also den in doppelte Anführungszeichen eingeschlossenen Dateipfad. Dieser Pfad wird entweder manuell oder über das Suchdialogfeld festgelegt. Dies ist die Standardeinstellung.
- Wenn die Scripteigenschaft `%%DieserWert%%` enthält und der Laufzeitwert des Textfelds leer ist, generiert das Steuerelement für die Dateisuche kein Script.

**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

**Dateitypfilter:** Im Dialogfeld "Dateifilter" können Sie die Dateitypen festlegen, die in den Dropdown-Listen "Dateityp" und "Speichern als Typ" für Dialogfelder zum Öffnen und Speichern von Dateien angezeigt werden, die über ein Steuerelement des Dateisystembrowsers geöffnet werden. Standardmäßig sind alle Dateitypen verfügbar.

So wählen Sie Dateitypen aus, die nicht ausdrücklich im Dialogfeld angegeben werden:

1. Wählen Sie "Andere" aus.
2. Geben Sie einen Namen für den Dateityp ein.
3. Geben Sie einen Dateityp in der Form `*.Suffix` ein, beispielsweise `*.xls`. Sie können mehrere Dateitypen durch ein Semikolon getrennt angeben.

## Tabulator

Das Steuerelement "Registerkarte" fügt dem Dialogfeld für Knoten eine Registerkarte hinzu. Der neuen Registerkarte kann ein anderes Steuerelement hinzugefügt werden. Das Steuerelement "Registerkarte" besitzt die folgenden Eigenschaften:

**ID.** Die eindeutige ID für das Steuerelement.

**Titel.** Der Titel der Registerkarte.

**Position.** Legt die Position der Registerkarte im Dialogfeld für Knoten relativ zu den anderen Registerkarten im Dialogfeld für Knoten fest.

**Script.** Legt das R-Script oder Python for Spark-Script fest, das während der Ausführung von diesem Steuerelement generiert und ausgeführt wird und in die Scriptvorlage eingefügt werden kann.

- Sie können ein beliebiges gültiges R-Script oder Python for Spark-Script angeben und für Zeilenumbrüche `\n` verwenden.
- Der Wert `%%DieserWert%%` generiert eine durch Leerzeichen getrennte Liste des R-Scripts oder des Python for Spark-Scripts, das von den einzelnen Steuerelementen auf der Registerkarte generiert wird, und zwar in der Reihenfolge, in der sie auf der Registerkarte angezeigt werden (von oben nach unten und von links nach rechts). Dies ist die Standardeinstellung.
- Wenn die Eigenschaft **Script** `%%DieserWert%%` enthält und von keinem der Steuerelemente auf der Registerkarte ein R-Script oder ein Python for Spark-Script generiert wird, generiert die Registerkarte als Ganzes kein Script.

**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

## Unterdialofeldschaltfläche

Das Steuerelement "Unterdialofeldschaltfläche" definiert eine Schaltfläche, über die ein Unterdialofeld geöffnet wird, und bietet Zugriff auf die Dialogfelderstellung für das Unterdialofeld. Die Unterdialofeldschaltfläche besitzt folgende Eigenschaften:

**ID.** Die eindeutige ID für das Steuerelement.

**Titel.** Der Text, der auf der Schaltfläche angezeigt wird.

**QuickInfo.** Optionaler QuickInfo-Text, der angezeigt wird, wenn der Benutzer die Maus über das Steuerelement bewegt.

**Unterdialofeld** Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um die Dialogfelderstellung für das Unterdialofeld zu öffnen. Sie können den Aufbau auch öffnen, indem Sie auf die Unterdialofeldschaltfläche doppelklicken.

**Mnemonisches Zeichen.** Ein optionales Zeichen im Titel, das als Tastenkombination für das Steuerelement verwendet werden kann. Das Zeichen wird im Titel unterstrichen. Die Tastenkombination wird aktiviert, indem Sie Alt + [mnemonisches Zeichen] drücken.

**Aktivierungsregel.** Gibt eine Regel an, die festlegt, wann das aktuelle Steuerelement aktiviert wird. Klicken Sie auf die Schaltfläche mit Auslassungspunkten (...), um das Dialogfeld zu öffnen und die Regel anzugeben. Die Eigenschaft **Aktivierungsregel** wird nur angezeigt, wenn andere Steuerelemente zur Angabe einer Aktivierungsregel im Erstellungsbereich vorhanden sind.

**Anmerkung:** Das Steuerelement "Unterdialofeldschaltfläche" kann nicht in Unterdialofelder integriert werden.

**Eigenschaften für Unterdialofelder:** So zeigen Sie Eigenschaften für Unterdialofelder an und definieren diese Eigenschaften:

1. Öffnen Sie das Unterdialofeld, indem Sie im Hauptdialogfeld auf die Schaltfläche für das Unterdialofeld doppelklicken oder einmal auf die Unterdialofeldschaltfläche und anschließend auf die Schaltfläche mit Auslassungspunkten (...) für die Unterdialofeldeigenschaft klicken.
2. Klicken Sie im Unterdialofeld außerhalb der Steuerelemente in den Erstellungsbereich. Wenn keine Steuerelemente im Erstellungsbereich enthalten sind, werden die Unterdialofeldeigenschaften immer angezeigt.

**Unterdialofeldname.** Die eindeutige ID für das Unterdialofeld. Die Eigenschaft "Unterdialofeldname" ist ein Pflichtfeld.

**Anmerkung:** Wenn Sie den Unterdialofeldnamen in der Scriptvorlage als ID - in der Form %%Name meines Unterdialofelds%% - angeben, wird er während der Ausführung durch eine durch Leerzeichen getrennte Liste des Scripts ersetzt, das von den einzelnen Steuerelementen im Unterdialofeld generiert wird, und zwar in der Reihenfolge, in der sie auftreten (von oben nach unten und von links nach rechts).

**Titel.** Legt den Text fest, der in der Titelleiste des Unterdialofelds angezeigt wird. Die Eigenschaft "Titel" ist optional, es wird jedoch empfohlen, sie einzurichten.

**Hilfdatei.** Legt den Pfad zu einer optionalen Hilfdatei für das Unterdialofeld fest. Diese Datei wird geöffnet, wenn der Benutzer im Unterdialofeld auf die Schaltfläche **Hilfe** klickt. Dabei kann es sich um

die Datei handeln, die auch für das Hauptdialogfeld angegeben wurde. Bei Hilfedateien muss es sich um HTML-Dateien handeln. Weitere Informationen finden Sie in der Beschreibung für die Eigenschaft "Hilfedatei" für Dialogfeldeigenschaften.

## **Angeben einer Aktivierungsregel für ein Steuerelement**

Sie können eine Regel angeben, die festlegt, wann ein Steuerelement aktiviert wird. Sie können beispielsweise angeben, dass eine Optionsgruppe aktiviert wird, wenn eine Feldauswahl gefüllt ist. Die verfügbaren Optionen für die Angabe der Aktivierungsregel hängen vom Typ des Steuerelements ab, das die Regel definiert.

### **Feldauswahl**

Sie können angeben, dass das aktuelle Steuerelement aktiviert wird, wenn eine Feldauswahl mit mindestens einem (nicht leeren) Feld gefüllt ist. Alternativ können Sie angeben, dass das aktuelle Steuerelement aktiviert wird, wenn eine Feldauswahl nicht gefüllt (leer) ist.

### **Kontrollkästchen oder Kontrollkästchengruppe**

Sie können angeben, dass das aktuelle Steuerelement aktiviert wird, wenn ein Kontrollkästchen oder eine Kontrollkästchengruppe aktiviert ist. Alternativ können Sie angeben, dass das aktuelle Steuerelement aktiviert wird, wenn ein Kontrollkästchen oder eine Kontrollkästchengruppe nicht aktiviert ist.

### **Kombinationsfeld oder Listenfeld mit Einzelauswahl**

Sie können angeben, dass das aktuelle Steuerelement aktiviert wird, wenn ein bestimmter Wert in einem Kombinationsfeld oder in einem Listenfeld mit Einzelauswahl ausgewählt ist. Alternativ können Sie angeben, dass das aktuelle Steuerelement aktiviert wird, wenn ein bestimmter Wert in einem Kombinationsfeld oder in einem Listenfeld mit Einzelauswahl nicht ausgewählt ist.

### **Listenfeld mit Mehrfachauswahl**

Sie können angeben, dass das aktuelle Steuerelement aktiviert wird, wenn ein bestimmter Wert in den ausgewählten Werten in einem Listenfeld mit Mehrfachauswahl enthalten ist. Alternativ können Sie angeben, dass das aktuelle Steuerelement aktiviert wird, wenn ein bestimmter Wert nicht in den ausgewählten Werten in einem Listenfeld mit Mehrfachauswahl enthalten ist.

### **Optionsgruppe**

Sie können angeben, dass das aktuelle Steuerelement aktiviert wird, wenn ein bestimmtes Optionsfeld ausgewählt ist. Alternativ können Sie angeben, dass das aktuelle Steuerelement aktiviert wird, wenn ein bestimmtes Optionsfeld nicht ausgewählt ist.

Steuerelemente, für die eine Aktivierungsregel angegeben werden kann, verfügen über eine zugeordnete Eigenschaft **Aktivierungsregel**.

### **Anmerkung:**

- Aktivierungsregeln werden angewendet, unabhängig davon, ob das Steuerelement, das die Regel angibt, aktiviert ist. Betrachten Sie beispielsweise eine Regel, die angibt, dass eine Optionsgruppe aktiviert wird, wenn eine Feldauswahl gefüllt ist. Die Optionsgruppe wird immer aktiviert, wenn die Feldauswahl gefüllt ist, unabhängig davon, ob die Feldauswahl aktiviert ist.
- Wenn ein Steuerelement "Registerkarte" inaktiviert wird, werden alle Steuerelemente auf der Registerkarte inaktiviert, unabhängig davon, ob diese Steuerelemente eine Aktivierungsregel aufweisen, die erfüllt wird.
- Wenn eine Kontrollkästchengruppe inaktiviert wird, werden alle Steuerelemente in der Gruppe inaktiviert, unabhängig davon, ob das Kontrollkästchen aktiviert ist.

## **Erweiterungseigenschaften**

Das Dialogfeld **Erweiterungseigenschaften** gibt Informationen zur aktuellen Erweiterung in der Dialogfelderstellung für Erweiterungen an, beispielsweise den Namen der Erweiterung und die Dateien in der Erweiterung.

- Alle benutzerdefinierten Dialogfelder für Knoten, die in der Dialogfelderstellung für Erweiterungen erstellt werden, sind Teil einer Erweiterung.
- Felder auf der Registerkarte **Erforderlich** im Dialogfeld **Erweiterungseigenschaften** müssen vor der Installation einer Erweiterung und der in ihr enthaltenen benutzerdefinierten Dialogfelder für Knoten angegeben werden.

Wählen Sie in den Menüs zur Dialogfelderstellung für Erweiterungen folgende Optionen aus, um die Eigenschaften für eine Erweiterung anzugeben:

**Erweiterung > Eigenschaften**

## Erforderliche Eigenschaften von Erweiterungen

**Name** Ein eindeutiger, der Erweiterung zugeordneter Name. Dieser kann aus bis zu drei Wörtern bestehen und es wird nicht zwischen Groß- und Kleinschreibung unterschieden. Der Name muss aus Zeichen im 7-Bit-ASCII-Code bestehen. Um mögliche Namenskonflikte zu vermeiden, können Sie einen aus mehreren Wörtern bestehenden Namen verwenden, wobei das erste Wort dem Namen Ihres Unternehmens entspricht oder aus einer URL besteht.

### Zusammenfassung

Eine kurze Beschreibung der Erweiterung, die in einer Zeile angezeigt werden soll.

### Version

Eine Versions-ID der Form x.x.x, wobei jede Komponente der ID eine Ganzzahl sein muss, z. B. 1.0.0. Wenn Sie keine Nullen angeben, werden diese automatisch impliziert. Die Versions-ID 3.1 impliziert beispielsweise 3.1.0. Die Versions-ID ist unabhängig von der IBM SPSS Modeler-Version.

### Mindestversion von SPSS Modeler

Die Mindestversion von SPSS Modeler, die erforderlich ist, um die Erweiterung auszuführen.

### Dateien

In der Liste **Dateien** werden die Dateien angezeigt, die derzeit in der Erweiterung enthalten sind. Klicken Sie auf **Hinzufügen**, um Dateien zu der Erweiterung hinzuzufügen. Sie können auch Dateien von der Erweiterung entfernen und Dateien in einen angegebenen Ordner extrahieren.

- Benutzerdefinierte Dialogfelder für Knoten haben den Dateityp `.cfe`.
- Übersetzungsdateien für Komponenten der Erweiterung werden über die Lokalisierungseinstellungen auf der Registerkarte **Optional** hinzugefügt.
- Sie können eine Readme-Datei zu der Erweiterung hinzufügen. Geben Sie den Dateinamen als `ReadMe.txt` an. Die Benutzer können über das Dialogfeld, in dem die Details für die Erweiterung angezeigt werden, auf die Readme-Datei zugreifen. Sie können auch lokalisierte Versionen der Readme-Datei einbeziehen. Geben Sie ihnen den Dateinamen `ReadMe_<Sprach-ID>.txt`, wie beispielsweise `ReadMe_de.txt` für eine Version in deutscher Sprache.

## Optionale Eigenschaften von Erweiterungen

### Allgemeine Eigenschaften

#### Beschreibung

Eine etwas detailliertere als die im Feld **Zusammenfassung** gegebene Beschreibung der Erweiterung. Sie können beispielsweise die wichtigsten Funktionen auflisten, die in der Erweiterung verfügbar sind.

#### Datum

Ein optionales Datum für die aktuelle Version der Erweiterung. Es wird keine Formatierung gestellt.

**Autor** Der Autor der Erweiterung. Sie können hier eine E-Mail-Adresse angeben.

## Verknüpfungen

Zur Erweiterung gehörende URLs, z. B. die Homepage des Autors. Das Format dieses Felds ist willkürlich, grenzen Sie daher mehrere URLs mit Leerzeichen, Kommas oder anderen sinnvollen Trennzeichen voneinander ab.

## Schlüsselwörter

Schlüsselwörter, denen die Erweiterung zugeordnet wird.

## Plattform

Informationen zu Einschränkungen, die für die Verwendung der Erweiterung auf bestimmten Betriebssystemplattformen gelten.

## Abhängigkeiten

### Maximale Version von SPSS Modeler

Die maximale Version von IBM SPSS Modeler, unter der die Erweiterung ausgeführt werden kann.

### Integration Plug-in for R erforderlich

Gibt an, ob Integration Plug-in for R erforderlich ist.

Wenn für die Erweiterung R-Pakete aus dem CRAN-Paket-Repository erforderlich sind, geben Sie die Namen dieser Pakete in das Steuerelement "Erforderliche R-Pakete" ein. Bei den Namen wird zwischen Groß- und Kleinschreibung unterschieden. Um das erste Paket hinzuzufügen, klicken Sie auf einen beliebigen Punkt im Steuerelement "Erforderliche R-Pakete", um das Eingabefeld zu markieren. Wenn sich der Cursor in einer bestimmten Zeile befindet und Sie die **Eingabetaste** drücken, wird eine neue Zeile erstellt. Sie können eine Zeile löschen, indem Sie sie auswählen und die Löschtaste drücken.

## Lokalisierung

### Benutzerdefinierte Knoten

Sie können übersetzte Versionen der properties-Datei (gibt alle im Dialogfeld für Knoten vorkommenden Zeichenfolgen an) für ein benutzerdefiniertes Dialogfeld für Knoten in der Erweiterung hinzufügen. Zum Hinzufügen von Übersetzungen für ein bestimmtes Dialogfeld für Knoten, wählen Sie das Dialogfeld aus, klicken Sie auf **Übersetzungen hinzufügen** und wählen Sie den Ordner aus, der die übersetzten Versionen enthält. Alle übersetzten Dateien für ein bestimmtes Dialogfeld für Knoten müssen sich in demselben Ordner befinden. Anweisungen zur Erstellung der Übersetzungsdateien finden Sie in „Erstellen lokalisierter Versionen benutzerdefinierter Dialogfelder für Knoten“ auf Seite 61.

### Ordner für Übersetzungskataloge

Sie können lokalisierte Versionen der Felder **Übersicht** und **Beschreibung** für die Erweiterung angeben, die angezeigt werden, wenn Endbenutzer die Details der Erweiterung über den Erweiterungshub anzeigen. Das Set aller lokalisierten Dateien für eine Erweiterung sollte sich im Ordner lang befinden. Navigieren Sie zum Ordner lang, der die lokalisierten Dateien enthält, und wählen Sie diesen Ordner aus.

Zur Angabe von lokalisierten Versionen der Felder **Übersicht** und **Beschreibung** erstellen Sie die Datei <Erweiterungsname>\_<Sprach-ID>.properties für jede Sprache, für die eine Übersetzung bereitgestellt wird. Wenn die Datei .properties für die aktuelle Benutzerschnittstellensprache während der Ausführung nicht gefunden wird, werden die Werte der auf den Registerkarten **Erforderlich** und **Optional** angegebenen Felder **Übersicht** und **Beschreibung** verwendet.

- <Erweiterungsname> ist der Wert des Felds **Name** für die Erweiterung, bei dem alle Leerzeichen durch Unterstriche ersetzt werden.
- <Sprach-ID> ist die ID für eine bestimmte Sprache. IDs für die von IBM SPSS Modeler unterstützten Sprachen werden im Folgenden angezeigt.

Die französischen Übersetzungen für die Erweiterung MYORG MYSTAT werden beispielsweise in der Datei MYORG\_MYSTAT\_fr.properties gespeichert.

Die Datei `.properties` muss die folgenden zwei Zeilen enthalten, die den lokalisierten Text für die zwei Felder angeben:

```
Summary=<lokalisierter Text für Feld "Übersicht">
Description=<lokalisierter Text für Feld "Beschreibung">
```

- Die Schlüsselwörter `Summary` und `Description` müssen in Englisch angegeben werden und der lokalisierte Text muss sich in derselben Zeile wie das Schlüsselwort befinden. Es dürfen keine Zeilenumbrüche verwendet werden.
- Die Datei muss die Codierung ISO 8859-1 aufweisen. Zeichen, die nicht direkt in dieser Codierung dargestellt werden können, müssen durch Unicode-Escapezeichen ("`\u`") entwertet werden.

Der Ordner `lang`, der die lokalisierten Dateien enthält, muss über den Unterordner `<Sprach-ID>` verfügen, der die lokalisierte Datei `.properties` für eine bestimmte Sprache enthält. Die französische Datei `.properties` muss sich beispielsweise im Ordner `lang/fr` befinden.

Sprach-IDs

**de.** Deutsch

**en.** Englisch

**es.** Spanisch

**fr.** Französisch

**it.** Italienisch

**ja.** Japanisch

**ko.** Koreanisch

**pl.** Polnisch

**pt\_BR.** Brasilianisches Portugiesisch

**ru.** Russisch

**zh\_CN.** Chinesisch vereinfacht

**zh\_TW.** Chinesisch traditionell

## Verwalten von benutzerdefinierten Dialogfeldern für Knoten

Mit der Dialogfelderstellung für Erweiterungen können Sie benutzerdefinierte Dialogfelder für Knoten verwalten, die von Ihnen oder anderen Benutzern erstellt wurden. Benutzerdefinierte Dialogfelder für Knoten müssen auf allen Instanzen von SPSS Modeler Client oder SPSS Modeler Batch, auf denen sie erforderlich sind, installiert werden, bevor sie verwendet werden können. Beachten Sie, dass für SPSS Modeler Server keine Installationen erforderlich sind, um ein benutzerdefiniertes Dialogfeld für Knoten im Servermodus zu verwenden.

**Anmerkung:** Nur in IBM SPSS Modeler erstellte benutzerdefinierte Dialogfelder für Knoten können geändert werden.

## Öffnen einer Erweiterung, die benutzerdefinierte Dialogfelder für Knoten enthält

Sie können eine Erweiterungsbundledatei (.mpe) öffnen, die die Spezifikationen für mindestens ein benutzerdefiniertes Dialogfeld für Knoten enthält, oder Sie können eine installierte Erweiterung öffnen. Sie können eines der Dialogfelder für Knoten in der Erweiterung ändern und die Erweiterung speichern oder installieren. Wenn Sie die Erweiterung installieren, werden auch die Dialogfelder für Knoten installiert, die in der Erweiterung enthalten sind. Wenn Sie die Erweiterung speichern, werden auch die Änderungen gespeichert, die an einem der Dialogfelder für Knoten in der Erweiterung vorgenommen wurden.

Wählen Sie in den Menüs zur Dialogfelderstellung für Erweiterungen folgende Optionen aus, um eine Erweiterungsbundledatei zu öffnen:

### Datei > Öffnen

Wählen Sie in den Menüs zur Dialogfelderstellung für Erweiterungen folgende Optionen aus, um eine installierte Erweiterung zu öffnen:

### Datei > Installierte öffnen

**Anmerkung:** Wenn Sie eine installierte Erweiterung öffnen und ändern, wird sie durch die Auswahl von **Datei > Installieren** erneut installiert, d. h., die vorhandene Version wird ersetzt. Wenn Sie **Bearbeiten** im Kontextmenü eines mit der Dialogfelderstellung erstellten Knotens auswählen, wird das Dialogfeld für Knoten nicht in der Dialogfelderstellung geöffnet.

## Speichern in einer Erweiterungsbundledatei

Wenn die in der Dialogfelderstellung für Erweiterungen geöffnete Erweiterung gespeichert wird, werden auch die benutzerdefinierten Dialogfelder für Knoten gespeichert, die in der Erweiterung enthalten sind. Erweiterungen werden in einer Erweiterungsbundledatei (.mpe) gespeichert.

Wählen Sie in den Menüs zur Dialogfelderstellung für Erweiterungen folgende Optionen aus:

### Datei > Speichern

## Installieren einer Erweiterung

Wenn Sie die in der Dialogfelderstellung für Erweiterungen geöffnete Erweiterung installieren, werden auch die benutzerdefinierten Dialogfelder für Knoten installiert, die in der Erweiterung enthalten sind. Bei der Installation einer vorhandenen Erweiterung wird die vorhandene Version ersetzt. Hierbei werden alle benutzerdefinierten Dialogfelder für Knoten in der Erweiterung ersetzt, die bereits installiert sind.

Wählen Sie in den Menüs zur Dialogfelderstellung für Erweiterungen folgende Optionen aus, um die derzeit geöffnete Erweiterung zu installieren:

### Datei > Installieren

Standardmäßig werden Erweiterungen an einer allgemeinen Position Ihres Betriebssystems installiert, auf die die Benutzer Schreibzugriff haben. Weitere Informationen finden Sie in „Installationsverzeichnisse für Erweiterungen“ auf Seite 29.

**Anmerkung:** In einem geöffneten Stream werden die vorhandenen Versionen der Dialogfelder für Knoten, die in der Erweiterung enthalten sind, nicht ersetzt. Wenn Sie einen Stream öffnen, der einen Dialogfelderstellungsknoten enthält, der neu installiert wurde, erhalten Sie eine Warnung.

## Deinstallieren einer Erweiterung

Wählen Sie in den Menüs zur Dialogfelderstellung für Erweiterungen folgende Optionen aus:

**Datei > Deinstallieren**

Wenn Sie eine Erweiterung deinstallieren, werden auch die Dialogfelder für Knoten deinstalliert, die in der Erweiterung enthalten sind. Sie können Erweiterungen auch über den Erweiterungshub deinstallieren.

## Importieren einer Paketdatei mit benutzerdefinierten Dialogfeldern

Sie können eine Paketdatei mit benutzerdefinierten Dialogfeldern (.cfd) in die Dialogfelderstellung für Erweiterungen importieren. Die .cfd-Datei wird in eine .cfe-Datei umgewandelt, die einer neuen Erweiterung hinzugefügt wird.

Wählen Sie in den Menüs zur Dialogfelderstellung für Erweiterungen folgende Optionen aus:

**Datei > Importieren**

Sie können .cfe-Dateien auch über das Dialogfeld **Erweiterungseigenschaften** zu einer Erweiterung hinzufügen. Auf dieses Dialogfeld können Sie über **Erweiterung > Eigenschaften** in der Dialogfelderstellung für Erweiterungen zugreifen.

## Hinzufügen eines benutzerdefinierten Dialogfelds für Knoten zu einer Erweiterung

Sie können ein neues benutzerdefiniertes Dialogfeld für Knoten zu einer Erweiterung hinzufügen.

Wählen Sie in den Menüs zur Dialogfelderstellung für Erweiterungen folgende Optionen aus:

**Erweiterung > Neues Dialogfeld**

## Wechseln zwischen mehreren benutzerdefinierten Dialogfeldern für Knoten in einer Erweiterung

Wenn die aktuelle Erweiterung mehrere benutzerdefinierte Dialogfelder für Knoten enthält, können Sie zwischen diesen wechseln.

Wählen Sie in den Menüs zur Dialogfelderstellung für Erweiterungen folgende Optionen aus:

**Erweiterung > Dialogfeld bearbeiten** und wählen Sie das benutzerdefinierte Dialogfeld für Knoten aus, das Sie verwenden wollen.

## Erstellen einer neuen Erweiterung

Wenn Sie in der Dialogfelderstellung für Erweiterungen eine neue Erweiterung erstellen, wird der Erweiterung ein neues leeres benutzerdefiniertes Dialogfeld für Knoten hinzugefügt.

Wählen Sie in den Menüs zur Dialogfelderstellung für Erweiterungen folgende Optionen aus, um eine neue Erweiterung zu erstellen:

**Datei > Neu**



## Erweiterungen in SPSS Modeler Batch oder in IBM SPSS Collaboration and Deployment Services

Wenn Sie Erweiterungen in einer Installation von SPSS Modeler Batch oder IBM SPSS Collaboration and Deployment Services verwenden wollen, stellen Sie sicher, dass die Umgebungsvariable *IBM\_SPSS\_MODELER\_EXTENSION\_PATH* in der Zielumgebung definiert ist und auf die Position mit den Erweiterungen verweist. Wenn die Streams, die einen benutzerdefinierten Knoten enthalten, in IBM SPSS Collaboration and Deployment Services Repository gespeichert wurden, bevor die Umgebungsvariable *IBM\_SPSS\_MODELER\_EXTENSION\_PATH* definiert wurde, müssen Sie die Streams erneut im Repository speichern, damit sie erfolgreich ausgeführt werden können.

**Anmerkung:** Stellen Sie sicher, dass die Version des Adapters von SPSS Modeler Batch oder IBM SPSS Collaboration and Deployment Services für SPSS Modeler mit der Version von SPSS Modeler Client übereinstimmt, in der die Erweiterung erstellt wurde.

## Erstellen lokalisierter Versionen benutzerdefinierter Dialogfelder für Knoten

Sie können lokalisierte Versionen von benutzerdefinierten Dialogfeldern für Knoten in allen Sprachen installieren, die von IBM SPSS Modeler unterstützt werden. Sie können jede Zeichenfolge lokalisieren, die in einem benutzerdefinierten Dialogfeld für Knoten enthalten ist. Auch die optionale Hilfedatei kann lokalisiert werden.

### Lokalisieren von Zeichenfolgen in Dialogfeldern

Sie müssen eine Kopie der properties-Datei für das benutzerdefinierte Dialogfeld für Knoten für jede Sprache erstellen, die Sie bereitstellen wollen. Die properties-Datei enthält sämtliche lokalisierbaren Zeichenfolgen für das Dialogfeld für Knoten.

Extrahieren Sie die Datei (.cfe) mit benutzerdefinierten Dialogfeldern für Knoten aus der Erweiterung, indem Sie die Datei im Dialogfeld **Erweiterungseigenschaften** (in der Dialogfelderstellung für Erweiterungen) auswählen und auf **Extrahieren** klicken. Extrahieren Sie anschließend den Inhalt der .cfe-Datei. Eine .cfe-Datei ist nichts anderes als eine ZIP-Datei. Der extrahierte Inhalt einer .cfe-Datei enthält eine properties-Datei für jede unterstützte Sprache. Dabei wird der Name der Datei für eine bestimmte Sprache durch `<Dialogfeldname>_<Sprach-ID>.properties` bezeichnet (siehe die Sprach-IDs in der Tabelle unten).

1. Öffnen Sie jede properties-Datei, die Sie übersetzen wollen, mit einem Texteditor, der UTF-8 unterstützt, unter Windows beispielsweise in Notepad. Ändern Sie die Werte für Eigenschaften, die lokalisiert werden müssen, jedoch nicht die Namen der Eigenschaften. Eigenschaften, die mit einem bestimmten Steuerelement verknüpft sind, erhalten ein Präfix mit der ID für das Steuerelement. Die Eigenschaft "QuickInfo" für ein Steuerelement mit der ID *options\_button* ist beispielsweise *options\_button\_quickinfo\_LABEL*. Titeleigenschaften werden einfach `<ID>_LABEL` benannt, zum Beispiel *options\_button\_LABEL*.
2. Fügen Sie die lokalisierten Versionen der properties-Dateien wieder über die Lokalisierungseinstellungen auf der Registerkarte **Optional** im Dialogfeld **Erweiterungseigenschaften** in die Datei (.cfe) mit benutzerdefinierten Dialogfeldern für Knoten ein. Weitere Informationen finden Sie in „Optionale Eigenschaften von Erweiterungen“ auf Seite 56.

Wenn das Dialogfeld für Knoten geöffnet wird, sucht IBM SPSS Modeler nach einer properties-Datei, deren Sprach-ID mit der aktuellen Sprache übereinstimmt, die über die Dropdown-Liste **Sprache** auf der Registerkarte **Allgemein** im Dialogfeld **Optionen** angegeben ist. Wenn keine passende properties-Datei gefunden wird, wird die Standarddatei `<Dialogfeldname>.properties` verwendet.

## Lokalisieren der Hilfedatei

1. Erstellen Sie eine Kopie der Hilfedatei für das benutzerdefinierte Dialogfeld für Knoten und lokalisieren Sie den Text in der gewünschten Sprache.
2. Benennen Sie die Kopie mit den in der Tabelle unten angegebenen Sprach-IDs in `<Hilfedatei>_<Sprach-ID>.htm` um. Wenn die Hilfedatei beispielsweise `myhelp.htm` heißt und Sie eine deutsche Version der Datei erstellen möchten, dann muss die lokalisierte Hilfedatei `myhelp_de.htm` benannt werden.

Speichern Sie alle lokalisierten Versionen der Hilfedatei in demselben Verzeichnis, in dem auch die nicht lokalisierte Version gespeichert wird. Wenn Sie die nicht lokalisierte Hilfedatei über die Eigenschaft **Hilfedatei** des Fensters **Dialogfeld - Eigenschaften** hinzufügen, werden die lokalisierten Versionen automatisch in das Dialogfeld für Knoten eingefügt.

Falls auch Begleitdateien wie Bilddateien lokalisiert werden müssen, müssen Sie die entsprechenden Pfade zu den lokalisierten Versionen in der Haupthilfedatei manuell ändern. Begleitdateien, einschließlich lokalisierter Versionen, müssen manuell in die Datei (`.cfe`) für benutzerdefinierte Dialogfelder für Knoten eingefügt werden. Informationen zum Zugriff auf Dateien mit benutzerdefinierten Dialogfeldern für Knoten sowie zu ihrer Änderung finden Sie im vorherigen Abschnitt *Lokalisieren von Zeichenfolgen in Dialogfeldern*.

Wenn das Dialogfeld für Knoten geöffnet wird, sucht IBM SPSS Modeler nach einer Hilfedatei, deren Sprach-ID mit der aktuellen Sprache übereinstimmt, die über die Dropdown-Liste **Sprache** auf der Registerkarte **Allgemein** im Dialogfeld **Optionen** angegeben ist. Wenn keine derartige Hilfedatei gefunden wird, wird die Hilfedatei verwendet, die für das Dialogfeld für Knoten angegeben wurde (die Datei, die in den Dialogfeldeigenschaften unter der Eigenschaft **Hilfedatei** angegeben ist).

Sprach-IDs

**de.** Deutsch

**en.** Englisch

**es.** Spanisch

**fr.** Französisch

**it.** Italienisch

**ja.** Japanisch

**ko.** Koreanisch

**pl.** Polnisch

**pt\_BR.** Brasilianisches Portugiesisch

**ru.** Russisch

**zh\_CN.** Chinesisch vereinfacht

**zh\_TW.** Chinesisch traditionell

*Hinweis:* Texte in benutzerdefinierten Dialogfeldern für Knoten und zugehörige Hilfedateien sind nicht auf die von IBM SPSS Modeler unterstützten Sprachen beschränkt. Sie können die Dialogfelder für Kno-

ten und den Hilfetext in einer beliebigen Sprache verfassen, ohne sprachspezifische Eigenschaften und Hilfedateien zu erstellen. Allen Benutzern Ihres Dialogfelds für Knoten wird der Text dann in dieser Sprache angezeigt.

## Importieren und Exportieren von Daten mithilfe von Python for Spark

Mit der Dialogfelderstellung für Erweiterungen können Sie benutzerdefinierte Knoten erstellen und Python for Spark-Skripts zum Lesen von Daten aus einer Datenquelle an einer beliebigen Position schreiben sowie Daten in einem beliebigen Format ausgeben, das von Apache Spark unterstützt wird.

Ein Benutzer möchte seine Daten beispielsweise in eine Datenbank schreiben. Er verwendet die Dialogfelderstellung für Erweiterungen und Python for Spark, um einen benutzerdefinierten JDBC-Exportknoten zu erstellen und führt dann das Modell zum Schreiben von Daten in eine Datenbank aus. Er kann außerdem einen benutzerdefinierten JDBC-Importknoten erstellen, um Daten aus der Datenbank zu lesen. Er könnte diese Methode auch verwenden, um Daten beispielsweise aus einer JSON-Datei in SPSS Modeler einzulesen. Nach dem Einlesen der Daten in SPSS Modeler kann er alle verfügbaren SPSS Modeler-Knoten verwenden, um sein Geschäftsproblem zu bearbeiten.

**Anmerkung:** Wenn Sie JDBC mit der Import- und Exportfunktionalität von Python for Spark verwenden wollen, müssen Sie Ihre JDBC-Treiberdatei in das Unterverzeichnis `as/lib` des IBM SPSS Modeler-Installationsverzeichnis kopieren.

### Importieren/Exportieren von Daten mithilfe von Python for Spark

1. Rufen Sie **Erweiterungen > Benutzerdefiniertes Dialogfelderstellungsprogramm**.
2. Wählen Sie unter den Dialogeigenschaften **Python for Spark** als Scripttyp und **Import** oder **Export** als Knotentyp aus.
3. Geben Sie andere Eigenschaften wie einen Dialognamen wie erforderlich ein.
4. Geben Sie das Python for Spark-Script zum Importieren oder Exportieren von Daten in den Scriptabschnitt ein oder fügen Sie es ein.
5. Klicken Sie auf **Installieren**, um das Python for Spark-Script zu installieren. Der Quellenpalette werden neue benutzerdefinierte Importknoten hinzugefügt und der Exportpalette werden neue benutzerdefinierte Exportknoten hinzugefügt.

## Importieren und Exportieren von Daten mithilfe von R

Mit der Dialogfelderstellung für Erweiterungen können Sie benutzerdefinierte Knoten erstellen und R-Skripts zum Lesen von Daten aus einer Datenquelle an einer beliebigen Position schreiben sowie Daten in einem beliebigen Format ausgeben, das von R unterstützt wird.

Ein Benutzer möchte seine Daten beispielsweise in eine Datenbank schreiben. Er verwendet die Dialogfelderstellung für Erweiterungen und R, um einen benutzerdefinierten JDBC-Exportknoten zu erstellen und führt dann das Modell zum Schreiben von Daten in eine Datenbank aus. Er kann außerdem einen benutzerdefinierten JDBC-Importknoten erstellen, um Daten aus der Datenbank zu lesen. Er könnte diese Methode auch verwenden, um Daten beispielsweise aus einer JSON-Datei in SPSS Modeler einzulesen. Nach dem Einlesen der Daten in SPSS Modeler kann er alle verfügbaren SPSS Modeler-Knoten verwenden, um sein Geschäftsproblem zu bearbeiten.

### Importieren/Exportieren von Daten mithilfe von R

1. Rufen Sie **Erweiterungen > Benutzerdefiniertes Dialogfelderstellungsprogramm**.
2. Wählen Sie unter den Dialogeigenschaften **R** als Scripttyp und **Import** oder **Export** als Knotentyp aus.
3. Geben Sie andere Eigenschaften wie einen Dialognamen wie erforderlich ein.
4. Geben Sie das R-Script zum Importieren oder Exportieren von Daten in den Scriptabschnitt ein oder fügen Sie es ein.

5. Klicken Sie auf **Installieren**, um das R-Script zu installieren. Der Quellenpalette werden neue benutzerdefinierte Importknoten hinzugefügt und der Exportpalette werden neue benutzerdefinierte Exportknoten hinzugefügt.

---

## Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden. IBM stellt dieses Material möglicherweise auch in anderen Sprachen zur Verfügung. Für den Zugriff auf das Material in einer anderen Sprache kann eine Kopie des Produkts oder der Produktversion in der jeweiligen Sprache erforderlich sein.

Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim zuständigen IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Services von IBM verwendet werden können. Anstelle der IBM Produkte, Programme oder Services können auch andere, ihnen äquivalente Produkte, Programme oder Services verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte von IBM verletzen. Die Verantwortung für den Betrieb von Produkten, Programmen und Services anderer Anbieter liegt beim Kunden.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

*IBM Director of Licensing  
IBM Europe, Middle East & Africa  
Tour Descartes  
2, avenue Gambetta  
92066 Paris La Defense  
France*

Diese Informationen können technische Ungenauigkeiten oder typografische Fehler enthalten. Die hier enthaltenen Informationen werden in regelmäßigen Zeitabständen aktualisiert und als Neuausgabe veröffentlicht. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Websites anderer Anbieter werden lediglich als Service für den Kunden bereitgestellt und stellen keinerlei Billigung des Inhalts dieser Websites dar. Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Websites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängig voneinander erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

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

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des in diesem Dokument beschriebenen Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt auf der Basis der IBM Rahmenvereinbarung bzw. der Allgemeinen Geschäftsbedingungen von IBM, der IBM Internationalen Nutzungsbedingungen für Programmpakete oder einer äquivalenten Vereinbarung.

Die angeführten Leistungsdaten und Kundenbeispiele dienen nur zur Illustration. Die tatsächlichen Ergebnisse beim Leistungsverhalten sind abhängig von der jeweiligen Konfiguration und den Betriebsbedingungen.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen. IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Aussagen über Pläne und Absichten von IBM unterliegen Änderungen oder können zurückgenommen werden und repräsentieren nur die Ziele von IBM.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufs. Sie sollen nur die Funktionen des Lizenzprogramms illustrieren und können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden; Ähnlichkeiten mit tatsächlichen Namen und Adressen sind rein zufällig.

---

## Marken

IBM, das IBM Logo und `ibm.com` sind Marken oder eingetragene Marken der IBM Corporation in den USA und/oder anderen Ländern. Weitere Produkt- und Servicennamen können Marken von IBM oder anderen Unternehmen sein. Eine aktuelle Liste der IBM Marken finden Sie auf der Webseite "Copyright and trademark information" unter [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Adobe, das Adobe-Logo, PostScript und das PostScript-Logo sind Marken oder eingetragene Marken der Adobe Systems Incorporated in den USA und/oder anderen Ländern.

Intel, das Intel-Logo, Intel Inside, das Intel Inside-Logo, Intel Centrino, das Intel Centrino-Logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium und Pentium sind Marken oder eingetragene Marken der Intel Corporation oder ihrer Tochtergesellschaften in den USA oder anderen Ländern.

Linux ist eine eingetragene Marke von Linus Torvalds in den USA und/oder anderen Ländern.

Microsoft, Windows, Windows NT und das Windows-Logo sind Marken von Microsoft Corporation in den USA und/oder anderen Ländern.

UNIX ist eine eingetragene Marke von The Open Group in den USA und anderen Ländern.

Java und alle auf Java basierenden Marken und Logos sind Marken oder eingetragene Marken der Oracle Corporation und/oder ihrer verbundenen Unternehmen.

---

## Bedingungen für Produktdokumentation

Die Berechtigungen zur Nutzung dieser Veröffentlichungen werden Ihnen auf der Basis der folgenden Bedingungen gewährt.

## **Anwendbarkeit**

Diese Bedingungen sind eine Ergänzung der Nutzungsbedingungen auf der IBM Website.

## **Persönliche Nutzung**

Sie dürfen diese Veröffentlichungen für Ihre persönliche, nicht kommerzielle Nutzung unter der Voraussetzung vervielfältigen, dass alle Eigentumsvermerke erhalten bleiben. Sie dürfen diese Veröffentlichungen oder Teile der Veröffentlichungen ohne ausdrückliche Genehmigung von IBM weder weitergeben oder anzeigen noch abgeleitete Werke davon erstellen.

## **Kommerzielle Nutzung**

Sie dürfen diese Veröffentlichungen nur innerhalb Ihres Unternehmens und unter der Voraussetzung, dass alle Eigentumsvermerke erhalten bleiben, vervielfältigen, weitergeben und anzeigen. Sie dürfen diese Veröffentlichungen oder Teile der Veröffentlichungen ohne ausdrückliche Genehmigung von IBM außerhalb Ihres Unternehmens weder vervielfältigen, weitergeben oder anzeigen noch abgeleitete Werke davon erstellen.

## **Berechtigungen**

Abgesehen von den hier gewährten Berechtigungen werden keine weiteren Berechtigungen, Lizenzen oder Rechte (veröffentlicht oder stillschweigend) in Bezug auf die Veröffentlichungen oder darin enthaltene Informationen, Daten, Software oder geistiges Eigentum gewährt.

IBM behält sich das Recht vor, die hierin gewährten Berechtigungen nach eigenem Ermessen zurückzuziehen, wenn sich die Nutzung der Veröffentlichungen für IBM als nachteilig erweist oder wenn die obigen Nutzungsbestimmungen nicht genau befolgt werden.

Sie dürfen diese Informationen nur in Übereinstimmung mit allen anwendbaren Gesetzen und Vorschriften, einschließlich aller US-amerikanischen Exportgesetze und Verordnungen, herunterladen und exportieren.

IBM übernimmt keine Gewährleistung für den Inhalt dieser Veröffentlichungen. Diese Veröffentlichungen werden auf der Grundlage des gegenwärtigen Zustands (auf "as-is"-Basis) und ohne eine ausdrückliche oder stillschweigende Gewährleistung für die Handelsüblichkeit, die Verwendungsfähigkeit für einen bestimmten Zweck oder die Freiheit von Rechten Dritter zur Verfügung gestellt.





---

# Index

## Sonderzeichen

.par-Datei 13  
.pim-Datei 13

## A

Apache Spark 1

## D

Diagrammausgabe (Registerkarte) 17  
  Erweiterungsmodellnugget 20  
Dialogfelderstellung  
  Aktivieren von Regeln 55  
  Dateisuche 52  
  Dateitypfilter 53  
  Dialogfeldeigenschaften 32  
  Elementgruppe, Steuerelement 49  
  Farbauswahl 46  
  Feldauswahl 35  
  Feldquelle 37  
  Filtern von Feldlisten 37  
  geschützter Text 44  
  Hilfdatei 32  
  Kombinationsfeld 38  
  Kontrollkästchen 37  
  Kontrollkästchengruppe 51  
  Layoutregeln 32  
  Listenelemente für Kombinationsfeld 39  
  Listenelemente für Listenfeld 39  
  Listenfeld 39  
  Lokalisierung von Dialogfeldern und Hilfdateien 61  
  Optionsgruppe 49  
  Optionsgruppen, Felder 50  
  Python for Spark-Datenexport 63  
  Python for Spark-Datenimport 63  
  R-Datenexport 63  
  R-Datenimport 63  
  Registerkarte 53  
  Scriptvorlage 33  
  Spalten für Tabellensteuerelemente 47  
  Steuerelement für Datum 43  
  Steuerelement für statischen Text 45  
  Steuerelement für Tabelle 46  
  Textsteuerelement 41  
  Unterdialogfeldeigenschaften 54  
  Unterdialogfeldschaltfläche 54  
  Vorschau 34  
  Zahlensteuerelement 42  
Dialogfelderstellung für Erweiterungen 30  
  Dialogfelder für Knoten in installierten Erweiterungen ändern 58  
  Erweiterungen mit Dialogfeldern für Knoten installieren 58  
  Erweiterungen mit Dialogfeldern für Knoten öffnen 58

Dialogfelderstellung für Erweiterungen (Forts.)  
  Erweiterungen mit Dialogfeldern für Knoten speichern 58  
  Erweiterungsbundledateien 58

## E

Erweiterungen 25  
  Aktualisierungen für Erweiterungen installieren 26  
  Erweiterungen entfernen 27  
  Erweiterungsdetails 28  
  installierte Erweiterungen anzeigen 27  
  neue Erweiterungen suchen und installieren 26  
Erweiterungsausgabebrowser 16, 17  
Erweiterungsausgabeknoten 15  
  Ausgabe (Registerkarte) 16  
  Konsolenausgabe (Registerkarte) 16  
  Syntax (Registerkarte) 15  
Erweiterungsbundles  
  Installation von Erweiterungsbundles 29  
Erweiterungsexportknoten 12  
  Konsolenausgabe (Registerkarte) 13  
Erweiterungsimportknoten 22  
  Konsolenausgabe (Registerkarte) 22  
Erweiterungsknoten 12  
Erweiterungsmodellknoten 17  
  Konsolenausgabe (Registerkarte) 18  
  Modelloptionen (Registerkarte) 18  
  Textausgabe (Registerkarte) 18  
Erweiterungsmodellnugget 18  
  Diagrammausgabe (Registerkarte) 20  
  Informationen zu 18  
  Konsolenausgabe (Registerkarte) 20  
  Modelloptionen (Registerkarte) 19  
  Syntax (Registerkarte) 18  
  Textausgabe (Registerkarte) 20  
Erweiterungstransformationsknoten 21  
  Konsolenausgabe (Registerkarte) 22

## I

IBM SPSS Modeler-Erweiterungen 1  
IBM SPSS Modeler-Erweiterungsknoten 12  
IBM SPSS Modeler Solution Publisher 13

## K

Konsolenausgabe (Registerkarte)  
  Erweiterungsmodellnugget 20

## P

Python for Spark 1  
  API 2, 4, 7, 8, 9, 10

## R

R  
  zulässige Syntax 1

## S

Syntax (Registerkarte)  
  Erweiterungsmodellnugget 18

## T

Textausgabe (Registerkarte) 17  
  Erweiterungsmodellnugget 20

## V

Veröffentlichen von Streams  
  IBM SPSS Modeler Solution Publisher 13

## Z

Zulässige R-Syntax 1





