



Procedimiento R de WPS

guía del usuario y referencia

Versión: 4.4.3

(c) 2022 World Programming, an Altair Company

www.worldprogramming.com

Tabla de contenidos

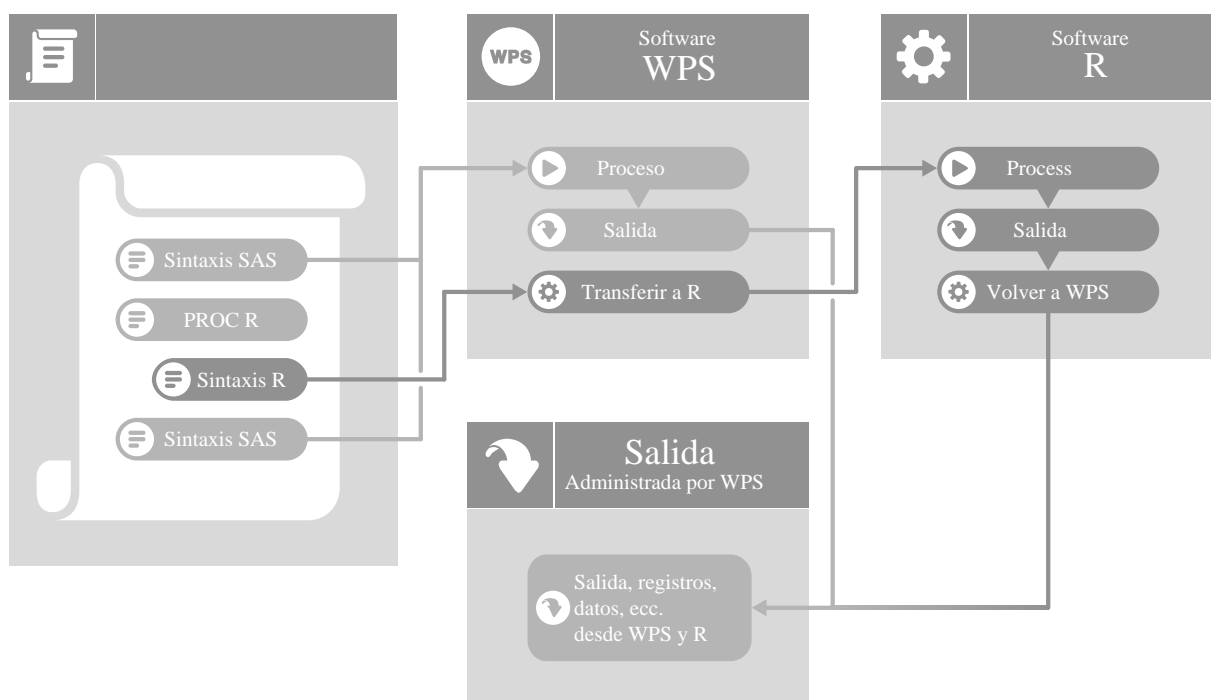
Procedimiento R.....	3
Introducción.....	4
Instalación y configuración.....	4
Instalación del intérprete R.....	5
Configuración de la variable de entorno <i>R_HOME</i>	6
Uso de R con WPS.....	6
Conversión del tipo de datos.....	7
Uso de gráficos de R.....	8
Uso de paquetes de R adicionales.....	9
Procesamiento de macro del lenguaje SAS.....	9
Ejemplo.....	10
Referencia del procedimiento R.....	11
PROC R.....	12
ASSIGN.....	14
ENDSUBMIT.....	16
EXECUTE.....	16
EXPORT.....	17
IMPORT.....	18
LOAD.....	19
SAVE.....	20
SUBMIT.....	22
Avisos legales.....	24

Procedimiento R

El procedimiento R permite que los programas WPS escritos en el lenguaje SAS incluyan el código escrito en el lenguaje R.

La combinación del lenguaje R y del lenguaje SAS permite que la mayor parte de una solución de análisis y procesamiento de datos se escriba en el lenguaje SAS de alto rendimiento y fuerza industrial, mientras que también utiliza las funciones presentes en el lenguaje R.

La siguiente imagen muestra cómo se procesa un programa en lenguaje SAS mediante el procedimiento R.



Introducción ↗	4
El procedimiento R permite a los programas de lenguaje SAS incluir código escrito en el lenguaje R.	
Instalación y configuración ↗	4
Configuración del entorno de R para WPS.	
Uso de R con WPS ↗	6
El uso de R en un programa en lenguaje SAS permite utilizar funciones que podrían no estar disponibles en WPS.	

Ejemplo ↗	10
Demuestra cómo usar un conjunto de datos de lenguaje SAS en el procedimiento R para crear un diagrama de dispersión.	
Referencia del procedimiento R ↗	11
Describe la sintaxis y las opciones para PROC R y sus instrucciones contenidas.	

Introducción

El procedimiento R permite a los programas de lenguaje SAS incluir código escrito en el lenguaje R.

Combinando R y el lenguaje SAS, puede:

- Utilizar el lenguaje SAS para procesar en masa y preparar datos, y pase los datos procesados a R.
- Utilizar los paquetes de R que ha desarrollado anteriormente para el análisis de datos.
- Utilizar paquetes o soluciones de análisis de datos de R que pueden no ser disponibles en el lenguaje SAS.

Recomendamos que los programas se codifiquen principalmente en el lenguaje de SAS, utilizando R donde se requieren estadísticas especializadas.

Los datos se transfieren entre el entorno de lenguaje SAS y el entorno de lenguaje lenguaje R mediante la instrucción `EXPORT`. Una vez que se han transferido los datos, están disponibles como un `data.frame` en un programa R. Al completar los datos del programa R, si es necesario, se pueden devolver al entorno de lenguaje SAS mediante la instrucción `IMPORT`.

Instalación y configuración

Configuración del entorno de R para WPS.

Cuando se usa R con WPS:

- El intérprete de R de 32 bits se requiere para la versión de 32 bits y el intérprete de R de 64 bits se requiere para la versión de WPS de 64 bits.
- No es necesario instalar módulos adicionales y no hay requisitos de licencia especiales para utilizar el procedimiento R.

El procedimiento R se puede usar con R versión 2.15.x y posterior, y actualmente es compatible con sistemas Microsoft Windows, Linux y macOS.

El software WPS no se envía con una copia de R. Para utilizar el procedimiento R, necesita una instalación separada de R.

Después de la instalación de R, configure la variable de entorno `R_HOME` para que apunte a la carpeta que contiene `libr.dll` en plataformas Windows o `libr.so` en plataformas UNIX o Linux.

Instalación del intérprete R

Plataformas de Windows

El paquete de instalación de Windows se puede descargar de <https://www.r-project.org/> e incluye versiones de R de 32 y 64 bits. El mismo instalador se puede utilizar para las versiones de 32 y 64 bits del software WPS.

De manera predeterminada, la instalación de R guarda la ubicación de instalación en el registro de Windows, que es donde WPS busca identificar la versión instalada actualmente. Esta es la versión de R instalada más recientemente y no se necesita una configuración especial de WPS para que WPS lo ubique.

Plataformas UNIX o Linux

WPS requiere la biblioteca compartida de R `libr.so` para interactuar con el intérprete de R. De manera predeterminada, esta biblioteca no se incluye con la distribución binaria R para plataformas UNIX. En plataformas UNIX o Linux, necesita compilar R desde el origen para incluir la biblioteca compartida requerida o instalar R utilizando el sistema de administración de paquetes de sistemas.

Para compilar desde el código fuente:

1. Necesita un conjunto mínimo de bibliotecas preinstaladas antes de poder compilar R desde el código fuente. Estos son equivalentes al paquete *build-essentials* más un JDK en Ubuntu.
2. Descargue el código fuente de R de <https://www.r-project.org/> y siga las instrucciones incluidas con la descarga. Asegúrese de usar la opción `--enable-R-shlib` cuando ejecute `configure` para construir la biblioteca compartida `libr.so`, por ejemplo:

```
./configure --enable-R-shlib --prefix=$HOME/R
```

Para más información, consulte la documentación de R en <https://cran.r-project.org/>.

Plataformas de macOS

La distribución binaria de R se puede instalar directamente desde el sitio web del proyecto R.

Para utilizar R con WPS, la variable `R_HOME` debe establecerse para apuntar al directorio de instalación que contiene la biblioteca compartida `libr.so`. La configuración predeterminada es: `/Library/Frameworks/Framework/Resources`. Debido a la forma en que se inician las aplicaciones en macOS, no es posible establecer `R_HOME` en un script de perfil de shell.

Para usar una versión específica de R, puede modificar la configuración apropiadamente, por ejemplo: `/Library/Frameworks/Framework/Versions/3.0/Resources`

Configuración de la variable de entorno *R_HOME*

Para ubicar la versión instalada de R, se debe establecer la variable de entorno *R_HOME*.

En plataformas Windows, la variable de entorno *R_HOME* debe apuntar a la carpeta que contiene el archivo `libr.dll`. En plataformas UNIX o Linux, la variable de entorno *R_HOME* debe apuntar a la carpeta que contiene el archivo `libr.so`.

Si está ejecutando WPS con R en una plataforma Unix o Linux, establezca la variable *R_HOME* para que apunte a la carpeta que contiene `libr.so`.

Si tiene varias instalaciones de R, configure la variable *R_HOME* como parte del programa de lenguaje SAS que ejecuta en WPS.

Establecer *R_HOME* antes de ejecutar WPS

La variable *R_HOME* se puede definir como una variable del sistema y utilizar con todas las aplicaciones de su dispositivo que ejecutan R. No es necesario establecer la variable en la plataforma Windows si se utiliza la ubicación de instalación de R predeterminada.

Establecer *R_HOME* en un programa de lenguaje SAS

La variable *R_HOME* se puede establecer en un programa de lenguaje SAS mediante la opción de sistema `SET`, por ejemplo:

```
OPTIONS SET = R_HOME 'C:\Program Files\R\R-3.5.0';
```

Esto establecerá la variable de entorno *R_HOME* durante la ejecución del programa de lenguaje SAS.

Uso de R con WPS

El uso de R en un programa en lenguaje SAS permite utilizar funciones que podrían no estar disponibles en WPS.

Los paquetes de R instalados se pueden importar y utilizar dentro del código R en línea utilizando la instrucción `library`, por ejemplo:

```
PROC R;  
  SUBMIT;  
    library(datasets)  
    data(iris)  
    summary(iris)  
    plot(iris)  
  ENDSUBMIT;  
RUN;
```

Cada uso posterior del procedimiento R en un programa de lenguaje SAS puede usar el mismo entorno R. Esto significa que cualquier variable global o paquete importado utilizado en una invocación del procedimiento R está disponible para todas las invocaciones del procedimiento R posteriores.

Cada invocación del procedimiento R puede incluir varios bloques de código de lenguaje R en la línea, y usar una combinación de código de lenguaje R en la línea, y los programas R se ejecutan mediante la instrucción EXECUTE.

Conversión del tipo de datos

Describe la correspondencia entre los formatos de lenguaje SAS y los tipos de datos de R.

Cuando se escriben datos en un `data.frame` utilizando la instrucción EXPORT del procedimiento R, los datos formateados se convierten en un tipo de datos R equivalente. WPS tiene muchos formatos que afectan la salida y la visualización de datos. Los formatos que sólo afectan el diseño de la salida de datos, como agregar símbolos de moneda o separadores de coma, no tienen efecto al escribir datos.

Se importa un `data.frame` mediante la instrucción IMPORT del procedimiento R. Cualquier objeto que se pueda forzar en un `data.frame` utilizando la función `as.data.frame` de R, se importa en el conjunto de datos de WPS.

Valores lógicos

Los valores lógicos se convierten en valores numéricos en el conjunto de datos de WPS. Los valores de los vectores de tipo lógico se convierten de la manera siguiente:

Valor de R	Valor de WPS
TRUE	1
FALSE	0

Valores enteros

El valor del lenguaje R NA, que se representa como el valor entero mínimo (-2147483648) se convierte en un valor ausente en lenguaje SAS.

Valores reales

Hay tres valores numéricos reales especiales en el lenguaje R:

- NA (No disponible). Representa un valor ausente.
- NaN (No es un número). Representa un valor indefinido, o un valor que no se puede mostrar, por ejemplo el resultado de cero dividido por cero.
- Inf (Infinito). Representan valores infinitos positivos y negativos. Por ejemplo, el resultado de intentar dividir cualquier valor entre cero.

Estos valores se convierten de la representación en lenguaje R a la representación en lenguaje SAS de la manera siguiente:

Valor de R	Valor de WPS
NA	. (valor ausente)
NaN	. (valor ausente)

Valor de R	Valor de WPS
+Inf	.I
-Inf	.M

Valores de fecha

Las variables enteras o reales en el lenguaje R que tienen una clase `Date` se formatean como `DATE9` . cuando se importan a WPS.

Los valores de `Date` en el lenguaje R representan un recuento de días desde la época Unix del 1 de enero de 1970 UTC. Las variables importadas se ajustan para tener en cuenta la época del lenguaje SAS del 1 de enero de 1960.

Valores de fecha y hora

Las variables reales en el lenguaje R que tienen una clase `POSIXct` se formatean como `DATETIME19` . cuando se importan a WPS.

Los valores de `POSIXct` en el lenguaje R representan un recuento de segundos desde la época Unix del 1 de enero de 1970 UTC. Las variables importadas se ajustan para tener en cuenta la época del lenguaje SAS del 1 de enero de 1960 y también utilizan el valor especificado en la opción `GMTOFFSET` de la instrucción `PROC R`.

Las variables reales en el lenguaje R que tienen una clase `times` se formatean como `TIME8` . cuando se importan a WPS.

Valores de carácter

En la importación, WPS analiza una variable de carácter y asigna un formato que es la longitud de la cadena más larga. Los valores individuales de una variable de carácter que son `NA` (No disponible) se convierten en el valor ausente de carácter en el lenguaje SAS (' ').

Valores de factores

Un *factor* en el lenguaje R es una forma de variable entera, donde los valores indexan una lista de variables categóricas (la lista se conoce como nivel en el lenguaje R). Cuando se importan en WPS, éstos se convierten en variables de caracteres en el conjunto de datos. La variable recibe una longitud igual a la cadena más larga de la lista de niveles.

Uso de gráficos de R

Al iniciar una sesión de R, WPS configura R así que los gráficos generados con el dispositivo gráfico predeterminado se capturen y se incluyan en la salida HTML ODS estándar de la sesión de WPS.

El siguiente programa amplía el ejemplo anterior incluyendo análisis y gráficos de regresión lineal simple.

1. Cree un nuevo archivo de programa, pegue el código siguiente y guarde el archivo:

```
data source;
  do x=1 to 10;
    y=ranuni(-1);
    output;
  end;

PROC R;
  export data=source;
  submit;
  model <- lm(source$y ~ source$x)
  print(model)
  par(mfrow=c(2, 2))
  plot(model)
endsubmit;
run;
```

2. Ejecute el programa haciendo clic en el icono **Ejecutar** de la barra de herramientas y examine la salida HTML.

La salida incluye los resultados de R impresos junto con un único gráfico generado en la sesión de R y enrutado a la salida de WPS.

```
Call:
lm(formula = source$y ~ source$x)
Coefficients:
(Intercept)      source$x
    0.5344         0.0241
```

Uso de paquetes de R adicionales

Para usar paquetes adicionales que no están incluidos en su instalación de R, le recomendamos que instale y verifique el funcionamiento básico de estos paquetes en el entorno interactivo de R. Los paquetes instalados se pueden usar en un programa R usando la función `library()`.

Un entorno de R iniciado por WPS hereda la lista de variables de entorno del proceso de WPS. Si se instala software de terceros para su uso en R que requiere, por ejemplo, entradas adicionales en la variable de entorno `PATH`, Workbench debe reiniciarse para registrar los cambios.

Procesamiento de macro del lenguaje SAS

Uso de macros del lenguaje SAS con programas R en línea.

Cuando se ejecuta un programa R en línea, el código entre las instrucciones `SUBMIT` y `ENDSUBMIT` se pasa literalmente al intérprete de R. Por lo tanto, el procesamiento de macros se suspende entre las instrucciones `SUBMIT` y `ENDSUBMIT` porque:

- El lenguaje R utiliza los caracteres & y % como parte de su sintaxis. El intento de procesar con macros el código fuente de R puede dar lugar a que la sintaxis válida de R sea malinterpretada como instrucciones de macros del lenguaje SAS.
- El lenguaje R permite comentarios de estilo de fin de línea. El contenido podría contener, por ejemplo, apóstrofes no emparejados; la conversión de la sintaxis R en tokens utilizando las reglas de análisis regular del lenguaje SAS no podría entonces ocurrir.

Ejemplo

Demuestra cómo usar un conjunto de datos de lenguaje SAS en el procedimiento R para crear un diagrama de dispersión.

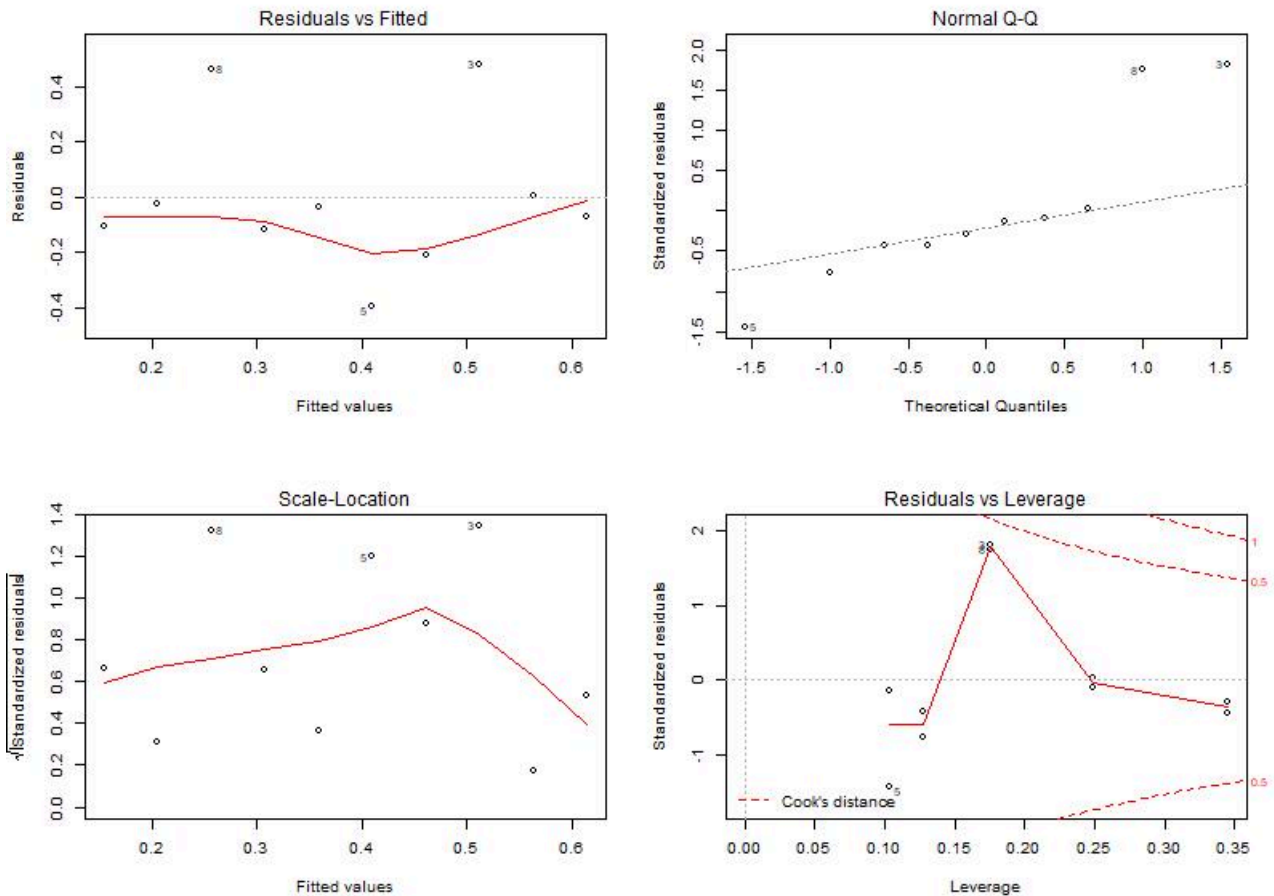
El siguiente ejemplo crea un conjunto de datos en un `DATA` step del lenguaje SAS y a continuación, usa la instrucción `EXPORT` para pasar ese conjunto de datos al entorno de R. El conjunto de datos se convierte en un `data.frame` como parte de la exportación y el `data.frame` se utiliza para crear gráficos en una cuadrícula de dos gráficos de ancho y dos de profundidad.

Se crea un destino de archivo PDF de salida mediante el Output Delivery System (ODS) del lenguaje SAS. La adición de PDF a los destinos de salida incluye el contenido de `data.frame` impreso y el archivo de imagen del diagrama devuelto en la salida PDF. El PDF se guarda y la salida se puede ver en un visor de PDF.

```
ODS PDF FILE='scatter_plot.pdf';
DATA SOURCE;
  DO X=1 TO 10;
    Y=RANUNI(-1);
    OUTPUT;
  END;
RUN;

PROC R;
  EXPORT DATA=source;
  SUBMIT;
    str(source)
    print(source)
    model <- lm(source$Y ~ source$X)
    print(model)
    par(mfrow=c(2, 2))
    plot(model)
    x <- (1:10)
  ENDSUBMIT;
  IMPORT R=x;
RUN;
ODS PDF CLOSE;
```

Esto crea el siguiente gráfico en la salida PDF de ODS:



Referencia del procedimiento R

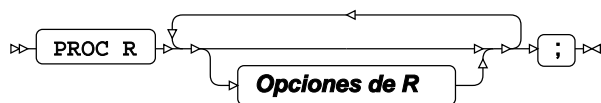
Describe la sintaxis y las opciones para PROC R y sus instrucciones contenidas.

PROC R ↗	12
Invoca el entorno de R que permite la ejecución de programas de lenguaje R en la línea o externos.	
ASSIGN ↗	14
La instrucción ASSIGN se puede utilizar para asignar valores de variables de lenguaje SAS a un vector R.	
ENDSUBMIT ↗	16
Especifica el final de un programa de lenguaje R en la línea.	
EXECUTE ↗	16
Ejecuta un programa R almacenado en un archivo separado.	

EXPORT ↗	17
Permite que un conjunto de datos de lenguaje SAS se convierta en un <code>data.frame</code> de R y se haga referencia a él en un programa R.	
IMPORT ↗	18
Permite que un <code>data.frame</code> de lenguaje R se convierta en un conjunto de datos de lenguaje SAS y se haga referencia en un programa de lenguaje SAS.	
LOAD ↗	19
La instrucción <code>LOAD</code> deserializa un <i>objeto de R</i> almacenado en un catálogo de lenguaje SAS.	
SAVE ↗	20
Permite serializar los objetos de R y almacenarlos en un catálogo lingüístico de SAS.	
SUBMIT ↗	22
Especifica el inicio de un programa de lenguaje lenguaje R en la línea.	

PROC R

Invoca el entorno de R que permite la ejecución de programas de lenguaje R en la línea o externos.



Los conjuntos de datos creados en WPS pueden estar disponibles para el programa R mediante la instrucción `EXPORT` y un conjunto de datos se puede importar desde el programa R a WPS mediante la instrucción `IMPORT`.

Un programa R puede escribirse en la línea en el procedimiento R o ejecutarse desde un archivo separado:

- Para ejecutar un programa R en la línea, use las instrucciones `SUBMIT` y `ENDSUBMIT`.
- Para ejecutar un programa R almacenado en un archivo externo, use la instrucción `EXECUTE`.

Se puede salir del entorno de entorno de R con una instrucción `RUN`.

Opciones

Las siguientes opciones están disponibles.

GMTOFFSET

Especifica el desplazamiento respecto a UTC que se aplica cuando se mueven conjuntos de datos entre los entornos de lenguaje SAS y lenguaje R para tener en cuenta la zona horaria actual.

```
➤ GMTOFFSET = "+/-HH:MM" ❏
```

Los valores de fecha y hora en el lenguaje R se representan en UTC (Coordinated Universal Time, Tiempo Universal Coordinado) con una zona horaria asociada. En el lenguaje SAS, los valores de fecha y hora no llevan implícita ninguna zona horaria. Se aplica el `GMTOFFSET` especificado cuando se usa la instrucción `ASSIGN`, `EXPORT` o `IMPORT`.

KEEP

Especifica que el entorno de R actual no se termina cuando finaliza el procedimiento.

Cuando se especifica, el entorno de R actual se mantiene activo cuando finaliza el procedimiento actual, y el entorno se utiliza en la siguiente invocación del procedimiento R en el mismo programa. Si esa invocación no especifica `KEEP`, el entorno finaliza cuando finaliza el procedimiento.

El comportamiento predeterminado es terminar el entorno de R al final del procedimiento. Especificar `KEEP` mantiene el entorno de R actual, incluidos los módulos cargados durante la ejecución de un programa R, que se utilizarán en la siguiente invocación de `PROC R`.

Puede especificar la opción del sistema `RKEEP` para usar el mismo entorno de R durante la ejecución del programa de lenguaje SAS.

LIB

Especifica la ubicación predeterminada de la biblioteca para el paso del procedimiento. La ubicación predeterminada es la biblioteca `WORK`.

Se utiliza la ubicación `LIB`:

- Cuando se exporta un conjunto de datos, y no se especifica *nombre-de-biblioteca* como parte de la ruta para la opción `DATA` de la instrucción `EXPORT`.
- Cuando se importa un conjunto de datos, y no se especifica *nombre-de-biblioteca* como parte de la ruta para la opción `DATA` de la instrucción `IMPORT`.
- Cuando se guarda un objeto R en un catálogo de lenguaje SAS, y no se especifica *nombre-de-biblioteca* como parte de la ruta para la opción `CATALOG` de la instrucción `SAVE`.
- Cuando se carga un objeto R en un catálogo de lenguaje SAS, y no se especifica *nombre-de-biblioteca* como parte de la ruta para la opción `CATALOG` de la instrucción `LOAD`.

TERMINATE

Especifica que el entorno de R se termina cuando finaliza el procedimiento.

Especificar `TERMINATE` detiene el entorno de R actual incluso si se ha especificado la opción de sistema `RKEEP`.

TIMESASCHRON

Especifica si los valores de tiempo están representados en R usando la clase `chron`.



De manera predeterminada, los valores de la hora se almacenan en el tipo `POSIXct` de R, que representa una cuenta de segundos desde la medianoche. Cuando se especifica `TIMESASCHRON`, los valores de tiempo se almacenan en R como tipos `chron.times`.

Para utilizar la opción `TIMESASCHRON`, debe incluir el paquete `chron` en su entorno R utilizando la instrucción `library()` de R.

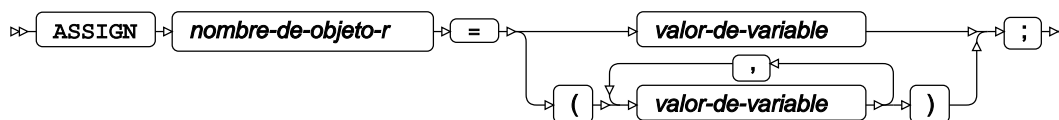
Ejemplo

El siguiente ejemplo muestra cómo utilizar `PROC R` para encontrar la versión del intérprete de R utilizado con WPS. La información de la versión se escribe en la salida ODS.

```
PROC R;
  SUBMIT;
    print(R.version)
  ENDSUBMIT;
RUN;
```

ASSIGN

La instrucción `ASSIGN` se puede utilizar para asignar valores de variables de lenguaje SAS a un vector R.



La instrucción `ASSIGN` se usa para pasar parámetros a un programa R, y el *nombre-de-objeto* especificado se puede usar en un programa en línea o en una ejecución de programa mediante el comando `EXECUTE`. El *nombre-de-objeto* distingue entre mayúsculas y minúsculas y, a diferencia de las variables de lenguaje SAS, se debe mencionar en el programa R utilizando las mismas mayúsculas y minúsculas que en la definición `ASSIGN`.

Las variables se pueden generar mediante la expansión o ejecución de variables de macro en lenguaje SAS. Esto permite preprocesar variables utilizando el lenguaje SAS y pasar el resultado a un programa R.

nombre-de-objeto-r

Especifica uno o más valores de variable para pasar a un programa R. El *nombre-de-objeto-r* especifica el nombre de la variable por la que se hace referencia al *valor-de-variable* en el programa R.

El *nombre-de-objeto* se puede definir mediante un literal de nombre de lenguaje SAS ('objeto-de-R.nombre'N) para crear un objeto de R que no sería válido en el lenguaje SAS. Por ejemplo, para asignar un valor *Peter* a un objeto de R *nombre.empleado*, la instrucción `ASSIGN` sería:

```
ASSIGN 'employee.firstname'N = 'Peter';
```

Se pueden asignar varios *valores-de-variables* a un sólo objeto de lenguaje R. En este caso, todos los valores deben ser del mismo tipo y estar en una lista separada por comas entre paréntesis.

Transferencia de múltiples variables a R

En este ejemplo, se transfieren varias variables vectoriales desde un programa de lenguaje SAS a un programa R que convierte los vectores en una trama de datos.

```
PROC R;  
  ASSIGN Nu = (1, 2, 3, 4, 5);  
  ASSIGN Ch = ('Cyan', 'Magenta', 'Yellow', 'Black', 'Green');  
  SUBMIT;  
    DFrame <- data.frame(Nu, Ch)  
    print (DFrame)  
  ENDSUBMIT;  
RUN;
```

Esto produce la siguiente salida:

```
Nu      Ch  
  1     Cyan  
  2 Magenta  
  3  Yellow  
  4   Black  
  5   Green
```

Asignación de una variable de macro de lenguaje SAS a un objeto de R

En este ejemplo, la variable de macro del lenguaje SAS *PARM* se transfiere a un programa R para determinar el tamaño de la muestra en una muestra generada aleatoriamente.

```
%LET PARM=15;  
PROC R;  
  ASSIGN parm=&PARM;  
  SUBMIT;  
    x<-sample(1:3, parm, replace=TRUE)  
    print(x);  
  ENDSUBMIT;  
RUN;
```

Esto produce la siguiente salida:

```
[1] 1 1 3 3 3 2 1 1 3 1 1 3 3 1 1
```

ENDSUBMIT

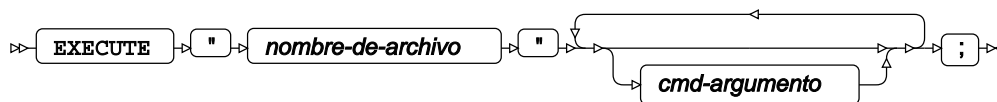
Especifica el final de un programa de lenguaje R en la línea.



La instrucción `ENDSUBMIT` debe ingresarse al comienzo de una nueva línea después del programa de lenguaje R.

EXECUTE

Ejecuta un programa R almacenado en un archivo separado.



El uso de la instrucción `EXECUTE` es una alternativa al uso de la instrucción `SUBMIT`. Permite que el código R se coloque en un archivo separado y le permite ejecutar el mismo programa tanto en WPS como en un entorno R interactivo.

nombre-de-archivo

Una cadena entre comillas que contiene la ruta del archivo de programa R. El *nombre-de-archivo* puede ser una ruta absoluta o una ruta relativa.

Cuando se usa Workbench para ejecutar un programa en lenguaje R, si se especifica una ruta relativa, la raíz de la ruta es el área de trabajo. Por ejemplo, para ejecutar un archivo denominado `math.r` desde un proyecto llamado `calculate`, la ruta relativa es `calculate/math.r`.

cmd-argumento

Especifica un argumento de línea de comando transferido al programa R.

Un ejemplo de ejecución de un programa R almacenado en un archivo

En este ejemplo, un programa R almacenado en un archivo externo `model.r` se ejecuta en el procedimiento R. El contenido del archivo de origen `model.r`:

```
model <- lm(source$Y ~ source$X)
print(model)
par(mfrow=c(2, 2))
plot(model)
```


El siguiente programa crea un conjunto de datos. El conjunto de datos se transfiere al programa R mediante la instrucción `EXPORT` antes de que el programa se ejecute usando el archivo `model.r`:

```
DATA SOURCE;
  DO X=1 TO 10;
    Y=RANUNI(-1);
    OUTPUT;
  END;

PROC R;
  EXPORT DATA=source;
  EXECUTE "model.r";
RUN;
```

EXPORT

Permite que un conjunto de datos de lenguaje SAS se convierta en un `data.frame` de R y se haga referencia a él en un programa R.

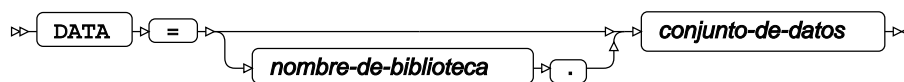


Opciones de exportación

Las siguientes opciones están disponibles con la instrucción `EXPORT`.

DATA

Especifica el nombre del conjunto de datos de WPS.



La ubicación de biblioteca se puede especificar con `nombre-de-biblioteca` en la opción `DATA` o la opción `LIB` de la instrucción `PROC R`:

- Si se especifica `nombre-de-biblioteca`, se usa siempre esa ubicación.
- Si se especifica la opción `LIB` de la instrucción `PROC R` y no se especifica `nombre-de-biblioteca`, se usa la ubicación en la opción `LIB`.
- Si no se especifica `nombre-de-biblioteca` ni la opción `LIB` en la instrucción `PROC R`, se usa la ubicación `WORK`.

R

Especifica el nombre de `data.frame` como se usa en el entorno de R.



Si esta opción no está especificada, el valor predeterminado del *dataframe-de-r* es el nombre del *conjunto-de-datos* especificado en la opción *DATA*. Si usa el nombre del *conjunto-de-datos* predeterminado en un programa de lenguaje R en la línea, el nombre de la variable debe coincidir con las mayúsculas y minúsculas utilizadas en la opción *DATA*.

Un ejemplo de exportación de datos de WPS a R

El siguiente ejemplo crea un conjunto de datos que contiene dos columnas numéricas. El conjunto de datos se exporta al entorno R y el contenido del `data.frame` se imprime en la salida ODS.

```
DATA SOURCE;
  DO X=1 TO 10;
    Y=RANUNI(-1);
  OUTPUT;
  END;

PROC R;
  EXPORT DATA=SOURCE;
  SUBMIT;
    str(source)
  ENDSUBMIT;
RUN;
```

Esto produce la siguiente salida:

```
'data.frame': 10 obs. of 2 variables:
 $ x: num  1 2 3 4 5 6 7 8 9 10
 $ y: num  0.371 0.924 0.59 0.434 0.962 ...
```

IMPORT

Permite que un `data.frame` de lenguaje R se convierta en un conjunto de datos de lenguaje SAS y se haga referencia en un programa de lenguaje SAS.

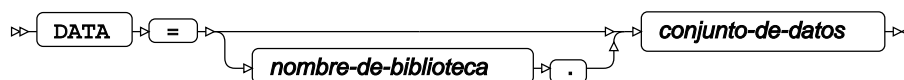


Opciones de importación

Las siguientes opciones están disponibles con la instrucción `IMPORT`.

DATA

Especifica la ubicación y el nombre del conjunto de datos como se usa en el entorno de lenguaje SAS de WPS Analytics.



La ubicación de biblioteca se puede especificar con *nombre-de-biblioteca* en la opción DATA o la opción LIB de la instrucción PROC R:

- Si se especifica *nombre-de-biblioteca*, se usa siempre esa ubicación.
- Si se especifica la opción LIB de la instrucción PROC R y no se especifica *nombre-de-biblioteca*, se usa la ubicación en la opción LIB.
- Si no se especifica *nombre-de-biblioteca* ni la opción LIB en la instrucción PROC R, se usa la ubicación WORK.

Si esta opción no está especificada, el valor predeterminado del *conjunto-de-datos* es el nombre del *dataframe-de-r* especificado en la opción R.

R

Especifica el nombre de `data.frame` como se usa en el entorno de R. Debe especificarse.



El *dataframe-de-r* distingue entre mayúsculas y minúsculas, y debe coincidir con las mayúsculas y minúsculas utilizadas para el `data.frame` importado en el programa R.

LOAD

La instrucción LOAD deserializa un *objeto de R* almacenado en un catálogo de lenguaje SAS.



Las instrucciones SAVE y LOAD permiten serializar un objeto de R y almacenarlo en un catálogo de lenguaje SAS, y deserializarlo posteriormente en una sesión de WPS. La instrucción SAVE serializa un objeto de R y lo almacena en una entrada de un catálogo. La instrucción LOAD deserializa un objeto de R de un catálogo.

Opciones de carga

Las siguientes opciones están disponibles con la instrucción LOAD.

CATALOG

Especifica el catálogo desde el que se carga el objeto de R almacenado.



Un catálogo de lenguaje SAS se define mediante:

nombre-de-biblioteca

Especifica el nombre de la biblioteca en la que se almacena el catálogo. La ubicación de biblioteca se puede especificar con *nombre-de-biblioteca* o la opción LIB de la instrucción PROC R.

- Si se especifica *nombre-de-biblioteca*, se usa siempre esa ubicación.
- Si se especifica la opción LIB de la instrucción PROC R y no se especifica *nombre-de-biblioteca*, se usa la ubicación en la opción LIB.
- Si no se especifica *nombre-de-biblioteca* ni la opción LIB en la instrucción PROC R, se usa la ubicación WORK.

catálogo

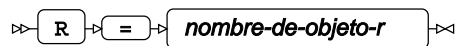
Especifica el nombre del catálogo.

entrada

Especifica el nombre del objeto de R en el catálogo.

R

Especifica el nombre de la variable para el objeto de R cargado tal y como se utiliza en el programa de lenguaje R.



Como R distingue entre mayúsculas y minúsculas, el *nombre-de-objeto-r* debe coincidir con las mayúsculas y minúsculas del nombre de la variable de R. El nombre puede especificarse utilizando la sintaxis de literal de nombre (por ejemplo, "r.objeto.nombre"N) si el nombre del objeto de R no sigue las reglas normales de los identificadores en el lenguaje SAS.

Un ejemplo de uso de la instrucción LOAD

```
proc r;
  load cat=catalog.entry r='target.object'n;
run;
```

SAVE

Permite serializar los objetos de R y almacenarlos en un catálogo lingüístico de SAS.



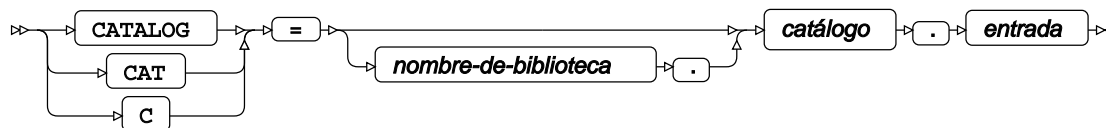
Las instrucciones SAVE y LOAD permiten serializar un objeto de R y almacenarlo en un catálogo de lenguaje SAS, y deserializarlo posteriormente en una sesión de WPS. La instrucción SAVE serializa un objeto de R y lo almacena en una entrada de un catálogo. La instrucción LOAD deserializa un objeto de R de un catálogo.

Opciones de guardar

Las siguientes opciones están disponibles con la instrucción `SAVE`.

CATALOG

Especifica la ubicación en la que se guardará el objeto de R.



La ubicación de biblioteca se puede especificar con *nombre-de-biblioteca* o la opción `LIB` de la instrucción `PROC R`:

nombre-de-biblioteca

Especifica el nombre de la biblioteca en la que se almacena el catálogo. La ubicación de biblioteca se puede especificar con *nombre-de-biblioteca* o la opción `LIB` de la instrucción `PROC R`.

- Si se especifica *nombre-de-biblioteca*, se usa siempre esa ubicación.
- Si se especifica la opción `LIB` de la instrucción `PROC R` y no se especifica *nombre-de-biblioteca*, se usa la ubicación en la opción `LIB`.
- Si no se especifica *nombre-de-biblioteca* ni la opción `LIB` en la instrucción `PROC R`, se usa la ubicación `WORK`.

catálogo

Especifica el nombre del catálogo.

entrada

Especifica el nombre del objeto de R en el catálogo.

R

Especifica el nombre del objeto de R para guardar en el catálogo. Debe especificarse.



Como R distingue entre mayúsculas y minúsculas, las mayúsculas y minúsculas de *nombre-de-objeto-r* debe coincidir con el nombre del objeto en el programa R. El nombre puede especificarse utilizando la sintaxis de literal de nombre (por ejemplo, "r.objeto.nombre"N) si el nombre del objeto de R no sigue las reglas normales de los identificadores en el lenguaje SAS.

DESCRIPTION

Especifica una cadena de descripción guardada con la entrada del catálogo.



La descripción se muestra en la salida de la instrucción `PROC CATALOG`.

La entrada del catálogo tendrá un tipo de `OBJECT`.

Un ejemplo de guardar un objeto de R en un catálogo de WPS

```
proc r;
  save cat=catalog.entry r='source.object'n;
run;
```

SUBMIT

Especifica el inicio de un programa de lenguaje R en la línea.



Un programa de lenguaje R en la línea se define como parte del procedimiento R en un programa de lenguaje SAS. La instrucción `SUBMIT` marca el inicio del programa y la instrucción `ENDSUBMIT` marca el final.

Un programa en lenguaje R debe comenzar en una nueva línea después de la instrucción `SUBMIT`, y la instrucción `ENDSUBMIT` debe aparecer al principio de una línea por sí sola.

Pueden existir varios programas de lenguaje R en línea en un único entorno del procedimiento R. Cada programa de lenguaje R se ejecuta a medida que se encuentra. Las variables definidas en un programa de lenguaje R se pueden usar en programas en la línea subsiguientes en el mismo procedimiento R.

Opciones SUBMIT

La siguiente opción se puede utilizar con la instrucción `SUBMIT`.

símbolo-r

Permite la sustitución de un símbolo en el programa R con una cadena.



Antes de pasar al entorno R, un programa R en línea se procesa previamente para reemplazar cualquier *símbolo-r* definido en el programa con el contenido del *valor-de-sustitución*. Si el *valor-de-sustitución* es una cadena, debe estar entre comillas. Si el *valor-de-sustitución* es una variable de macro de lenguaje SAS, se le debe anteponer una Y comercial (&), pero las comillas no son obligatorias.

El nombre del *símbolo-r* distingue entre mayúsculas y minúsculas y se debe referir a él en el programa R utilizando el mismo caso que en la definición de la instrucción `SUBMIT`.

```
PROC R;
SUBMIT greeting = 'Hello World';
r.welcome <- "&greeting."
print (r.welcome)
ENDSUBMIT;
RUN;
```

La sintaxis para sustituir un *símbolo-r* definido es la misma que la utilizada para la sustitución de variables de macros en lenguaje SAS, se debe anteponer una Y comercial (&) en el código en línea en el nombre del *símbolo-r* definido .

Instrucción SUBMIT – ejemplo básico

El siguiente ejemplo muestra cómo incorporar un programa en lenguaje R a un programa en lenguaje SAS:

```
PROC R;  
SUBMIT;  
x <- (1:10)  
print(x)  
ENDSUBMIT;  
RUN;
```

que produce las siguientes salidas:

```
[1] 1 2 3 4 5 6 7 8 9 10
```

Instrucción SUBMIT – uso de una variable de macro en un programa R

El siguiente ejemplo muestra cómo pasar una variable de un programa en lenguaje SAS a un programa en lenguaje R. La variable *bienvenido* se define mediante la macro del lenguaje SAS %LET. La variable de macro especificada se asigna al *símbolo-r* de la instrucción SUBMIT. El *símbolo-r* se referencia entonces en el programa en línea del lenguaje R:

```
%LET welcome = 'Hello World';  
PROC R;  
SUBMIT greeting = &welcome;  
r.welcome <- "&greeting"  
print (toupper(r.welcome))  
ENDSUBMIT;  
RUN;
```

que produce las siguientes salidas:

```
[1] "HELLO WORLD"
```

Avisos legales

(c) 2022 World Programming, an Altair Company

La presente información es confidencial y está sujeta a derecho de autor. Ninguna parte de esta publicación se puede reproducir o transmitir de ninguna forma, ni por ningún medio, ya sea electrónico o mecánico, incluyendo fotocopia, grabación o por cualquier sistema de almacenamiento y recuperación de información.

Marcas comerciales

WPS e World Programming son marcas registradas o comerciales de World Programming Limited en la Unión Europea y en otros países. (r) o ® indican una marca comunitaria.

SAS y todos los otros nombres de productos o servicios de SAS Institute Inc. son marcas registradas o comerciales de SAS Institute Inc. en los EE.UU. y en otros países. ® indica la registración en los EE.UU.

Todas las otras marcas comerciales mencionadas pertenecen a sus respectivos propietarios.

Avisos generales

World Programming Limited no está asociada de ninguna manera con SAS Institute Inc.

WPS no es SAS System.

Las expresiones "SAS", "lenguaje SAS" y "lenguaje de SAS" utilizadas en este documento, se usan en referencia al lenguaje de programación, llamado a menudo en una de dichas maneras.

Las expresiones "programa", "programa SAS" y "programa en el lenguaje SAS" utilizadas en este documento, se usan en referencia a los programas escritos en el lenguaje SAS. También se conocen como "scripts", "scripts SAS" o "scripts en el lenguaje SAS".

Las expresiones "IML", "lenguaje IML", "sintaxis IML", "Interactive Matrix Language" y "lenguaje de IML" utilizadas en este documento, se usan en referencia al lenguaje de programación, llamado a menudo en una de dichas maneras.

WPS incluye software desarrollado por terceros. Se puede encontrar más información en el archivo THANKS o acknowledgments.txt, incluidos en la instalación de WPS.