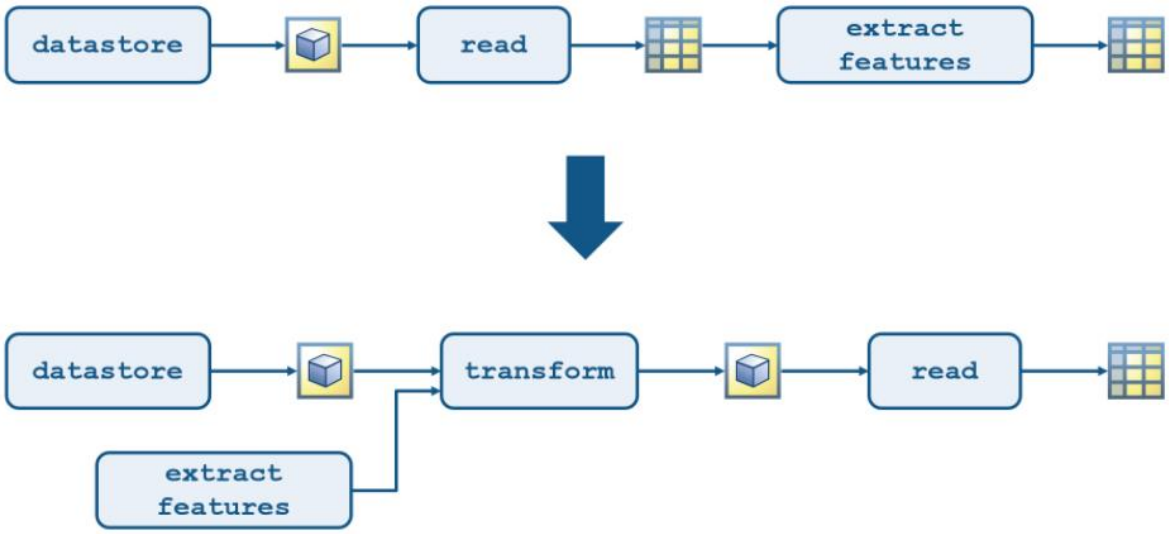
AUTOMATIZAR LA EXTRACCIÓN DE CARACTERÍSTICAS

Extraer Características de varios Archivos de Datos

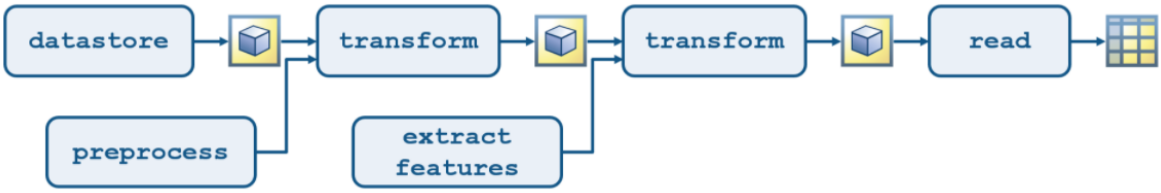
Almacenes de datos transformados

Para automatizar la extracción de características, le interesa que su almacén de datos aplique la función de extracción siempre que se lean los datos. Al igual que con el preprocesamiento, esto se puede hacer con un almacén de datos transformado.



Actividad 1

A partir de los datos sin procesar, normalmente tendrá que aplicar tanto las funciones de preprocesamiento como las de extracción de características. Puede aplicar la función `transform` repetidamente para agregar cualquier número de transformaciones del almacén de datos a los datos sin procesar.



El siguiente script aplica la función `escala` a los archivos del almacén de datos `ds_letras`. El almacén de datos transformado se almacena en la variable `ds_trans`.

```
ds_letras = datastore("*.txt")
ds_trans = transform(ds_letras,@escala)

function letra = escala(letra)
% Normaliza tiempo [0 1]
letra.Time = (letra.Time - letra.Time(1))/...
    (letra.Time(end) - letra.Time(1));
```

```
% Ajusta relación de aspecto
letra.X = 1.5*letra.X;
% Centra X & Y en (0,0)
letra.X = letra.X - mean(letra.X,"omitnan");
letra.Y = letra.Y - mean(letra.Y,"omitnan");
% Escala para tener un area = 1
scl = 1/sqrt(range(letra.X)*range(letra.Y));
letra.X = scl*letra.X;
letra.Y = scl*letra.Y;
end
```

Tarea: Utilice la función `transform` para aplicar la función `extrae` al almacén de datos `ds_trans`. Almacene el resultado en una variable llamada `caract_ds`.

```
function caract = extrae(letra)
% Relacion de aspecto
rel_asp = range(letra.Y)/range(letra.X);
% Numero Max/Min
idx_min = islocalmin(letra.X,"MinProminence",0.001);
num_X_min = nnz(idx_min);
idx_max = islocalmax(letra.Y,"MinProminence",0.001);
num_Y_max = nnz(idx_max);
% Velocidad
dT = diff(letra.Time);
dXdT = diff(letra.X)./dT;
dYdT = diff(letra.Y)./dT;
Vx = mean(dXdT,"omitnan");
Vy = mean(dYdT,"omitnan");
% Correlación
corr_XY = corr(letra.X,letra.Y,"rows","complete");
% Pone todo en una tabla
nom_caract = ["RelAspecto","NumXMin","NumYMax",...
    "Vx","Vy","CorrXY"];
caract = table(rel_asp,num_X_min,num_Y_max,Vx,Vy,...
    corr_XY,'VariableNames',nom_caract)
end
```

Actividad 2

Tarea: Utilice la función `readall` para leer, preprocesar y extraer características de todos los archivos de datos. Almacene el resultado en una variable llamada `datos`.

Hay 12 archivos y la función `extrae` calcula seis características para cada uno. Por lo tanto, `datos` debe ser una tabla de 12 por 6.

Visualice los datos importados creando un diagrama de dispersión de `RelAspecto` en el eje x y `CorrXY` en el eje y.

Actividad 3

Las letras que representan los datos se indican en los nombres de los archivos de datos, que tienen el formato `user`*`nnn_X_n`*`.txt`. Observe que el nombre de la letra aparece entre guiones bajos (*`_X_`*).

Puede usar la función `extractBetween` para extraer el texto situado entre las cadenas dadas.

```
extractedtxt = extractBetween(txt,"abc","xyz")
```

Si `txt` es el arreglo de cadenas `["hello abc 123 xyz","abcxyz","xyzabchelloxyzabc"]`, `extractedtxt` será `[" 123 ","","hello"]`.

Tarea: Utilice la función `extractBetween` para obtener los nombres de letras conocidos de los nombres de los archivos buscando el texto entre dos guiones bajos (`_`). Almacene el resultado en una variable llamada `letra_conocida`. Recuerde que los nombres de archivo se almacenan en la propiedad `Files` del almacén de datos `ds_letras`.

Actividad 4

En los problemas de clasificación, normalmente interesa representar la etiqueta conocida como una variable categórica. Puede utilizar la función `categorical` para convertir un arreglo al tipo categórico.

```
xcat = categorical(x)
```

De forma predeterminada, se utilizarán los valores únicos de `x` para definir el conjunto de categorías.

Tarea: Utilice la función `categorical` para convertir `letra_conocida` en categórico.

Actividad 5

Es conveniente tener las clases conocidas asociadas a los datos de entrenamiento. Recuerde que puede crear nuevas variables en una tabla mediante la asignación a una variable usando la notación de puntos.

```
T.newvar = workspacevar
```

Tarea: Agregue `letra_conocida` a la tabla `datos` como una nueva variable llamada `Caracter`.

Utilice la función `gscatter` para crear un diagrama de dispersión agrupado de `RelAspecto` en el eje x y `CorrXY` en el eje y, agrupado por `Caracter`.

Tarea adicional

Intente modificar `extrae` para cambiar las características que se calculan a partir de los datos. Compruebe que puede volver a ejecutar el script para obtener una nueva versión de la tabla `datos`.

Archivos requeridos:

`user001_M_1.txt`

`user001_V_1.txt`

`user002_M_1.txt`

`user002_V_1.txt`

`user003_M_1.txt`

`user003_M_2.txt`

`user003_V_1.txt`

`user003_V_1.txt`

`user004_M_1.txt`

`user004_V_1.txt`

`user005_M_1.txt`

`user005_V_1.txt`