Using Sweave and **patchDVI** with Japanese text

Duncan Murdoch

2020 年 5 月 4 日

概 要

The **patchDVI** package works with Sweave [**?**] and document previewers to facilitate editing: it modifies the links that LATEX puts into the output so that they refer to the original source. It also includes a few project management functions to make large multi-file Sweave documents easier to handle. This document describes Japanese language support in the package.

目 次

1 Introduction

The main vignette "The patchDVI package" describes general features of patchDVI, and how to set up various editors to use it in typesetting English-language vignettes. This vignette describes how to set it up for Japanese language text. I expect the issues would be the same in other non-Roman alphabets, but I have no experience with them.

The main issues in writing Japanese language (日本語) documents are the encoding of the character set and the varying support by different LaTeX engines. The standard `pdflatex` engine does not work well with Japanese. In November, 2013 I visited the Institute for Statistical Mathematics in Tokyo, and with the help of Mr. Yasuto Nakano and Prof. Haruhiko Okumura I developed these instructions. In outline: you must use the UTF-8 encoding for the characters, use `uplatex`, `platex` or `xelatex` to convert to a DVI file, and use `dvipdfmx` to convert the DVI to PDF. These instructions describe the procedure using `uplatex`; I expect modifications for the other engines will be minor.

2 Creating a Vignette in an R Package

To create a "vignette" in an R package that contains Japanese language text, follow these steps.

1. In the `DESCRIPTION` file of the package, add the lines

   ```
   Suggests:  patchDVI
   VignetteBuilder:  patchDVI
   ```

2. Put the Sweave file containing the source for the vignette in the `vignettes` directory of the package.

3. As with all vignettes, it needs a line like

   ```
   %\VignetteIndexEntry{Using Sweave and patchDVI with Japanese text}
   ```

   somewhere near the beginning, to give the text that will be shown by `browseVignettes()` and similar R functions.

4. To process it through `uplatex` and `dvipdfmx` instead of the default engines, add the line

   ```
   %\VignetteEngine{patchDVI::JSweave}
   ```

   near the beginning.

3 Using patchDVI with TeXShop

These instructions have been tested with TeXShop version 2.47 and 3.36.1, but should work with other versions as well.

1. In Preferences – Typesetting – Sync Method, choose "Sync-TeX".

2. Create a file called `Library/TeXShop/Engines/JSweave.engine` containing the two lines

   ```
   #!/bin/sh
   ```

   and

   ```
   Rscript -e "Sys.setlocale(locale='en_US.UTF-8');
   patchDVI::SweaveDVIPDFM(  '$1', latex='uplatex',
   dvipdfm='dvipdfmx', encoding = 'UTF-8' )"
   ```

   (which I have split to fit in this document; the actual line should not be split). Give this file executable permissions.

3. Install the `patchDVI` package into R. For this to work, you need at least version 1.9.1587 of patchDVI.

4. Add the lines

   ```
   %!TEX encoding = UTF-8 Unicode
   %!TEX TS-program = JSweave
   ```

   at the beginning of all files, and make sure they are created with UTF-8 encoding.

5. If you have multiple files in your project, your main file must be a Sweave file (e.g. `Main.Rnw`) which lists all Sweave files in a `.SweaveFiles` variable, and you need to add the line

   ```
   %!TEX root = Main.Rnw
   ```

   to each subordinate file.

6. Add

   ```
   \usepackage[utf8]{inputenc}
   ```

   to the main file to declare to Sweave that it uses the UTF-8 encoding. Add

   ```
   \newcommand{\ThisIsNotUsed}{
   \usepackage[utf8]{inputenc}
   }
   ```

to subordinate files. This defines a macro which is never used; the purpose is for Sweave to see the `inputenc` line. (You can also use this line in the main file if you are using a class file such as `ujarticle` which defines the encoding, and don't want `inputenc` to be used.)

7. Add the lines

```
\usepackage{Sweave}
\SweaveOpts {concordance=TRUE,eps=TRUE,pdf=FALSE}
```

near the start of each file. In subordinate files, the first line should be commented out, e.g.

```
% \usepackage{Sweave}
\SweaveOpts {concordance=TRUE,eps=TRUE,pdf=FALSE}
```

The reason for this is that Sweave generates `\inputgraphics` calls with no extensions on the files, and the `uplatex` processor will not find PDF files unless they are explicitly named. (Advanced users will know other ways to handle this issue.)

With these changes, the TeXShop previewer will support Sync-TeX; you right click in the preview, and choose Sync from the menu to jump to your source location.

4   Using patchDVI with TeXWorks

TeXWorks is an editor for several platforms, somewhat similar to TeXShop. These instructions have been tested in TeXWorks 0.4.4 on MacOS; I have not tested them on Windows, because I do not have `uplatex` and `dvipdfmx` there, but they should work if those programs are available.

1. In Edit – Preferences – Typesetting click on the "+" sign near the bottom of the Processing Tools box. Set the name of the tool to be JSweave. Set the program to Rscript. (You may need to give the fully qualified path if R is not in your system PATH.)

   Add two arguments on separate lines. The first is simply -e, the second is

```
{Sys.setlocale(locale='en_US.UTF-8');
 patchDVI::SweaveDVIPDFM(  '$fullname',
 latex='uplatex', dvipdfm='dvipdfmx',
 encoding = 'UTF-8' )}
```

This should be typed all on one line.

2. Install the `patchDVI` package into R. For this to work, you need at least version 1.9.1587 of patchDVI.

3. Add the lines

```
%!TEX encoding = UTF-8 Unicode
%!TEX program = JSweave
```

at the beginning of all files, and make sure they are created with UTF-8 encoding.

4. If you have multiple files in your project, your main file must be a Sweave file (e.g. `Main.Rnw`) which lists all Sweave files in a `.SweaveFiles` variable, and you need to add the line

```
%!TEX root = Main.Rnw
```

to each subordinate file.

5. Add

```
\usepackage[utf8]{inputenc}
```

to the main file to declare to Sweave that it uses the UTF-8 encoding. Add

```
\newcommand{\ThisIsNotUsed}{
\usepackage[utf8]{inputenc}
}
```

to subordinate files. This defines a macro which is never used; the purpose is for Sweave to see the `inputenc` line. (You can also use this line in the main file if you are using a class file such as `ujarticle` which defines the encoding, and don't want `inputenc` to be used.)

6. Add

```
 \SweaveOpts {concordance=TRUE,eps=TRUE,pdf=FALSE}
```

See the note in Section 3 point 7 for a discussion of this.

The TeXWorks previewer will jump back to the source if you right click and choose Jump to Source.