PDF Documents

- Overview
- Table of Contents
- Figure Options
- Syntax Highlighting
- LaTeX Options
- Advanced Customization
- Shared Options

Overview

To create a PDF document from R Markdown you specify the `pdf_document` output format in the front-matter of your document:

```
---
title: "Habits"
author: John Doe
date: March 22, 2005
output: pdf_document
---
```

Within R Markdown documents that generate PDF output you can use raw LaTeX and even define LaTeX macros. See the documentation on Raw TeX (authoring_pandoc_markdown.html#raw-tex) for details.

Table of Contents

You can add a table of contents using the `toc` option and specify the depth of headers that it applies to using the `toc_depth` option. For example:

```
---
title: "Habits"
output:
  pdf_document:
    toc: true
    toc_depth: 2
---
```

If the table of contents depth isn't explicitly specified then it defaults to 3 (meaning that all level 1, 2, and 3 headers will be included in the table of contents).

You can add section numbering to headers using the `number_sections` option:

```
---
title: "Habits"
output:
  pdf_document:
    toc: true
    number_sections: true
---
```

Figure Options

There are a number of options that affect the output of figures within PD documents:

- `fig_width` and `fig_height` can be used to control the default figure width and height (6 x 4.5 is used by default)
- `fig_crop` controls whether the the pdfcrop utility (if available) is automatically applied to pdf figures (this is true by default).
- `fig_caption` controls whether figures are rendered with captions (this is false by default).

For example:

```
---
title: "Habits"
output:
  pdf_document:
    fig_width: 7
    fig_height: 6
    fig_caption: true
---
```

Syntax Highlighting

The `highlight` option specifies the syntax highlighting style. Supported styles include “default”, “tango”, “pygments”, “kate”, “monochrome”, “espresso”, “zenburn”, and “haddock” (specify null to prevent syntax highlighting):

For example:

```
---
title: "Habits"
output:
  pdf_document:
    highlight: tango
---
```

LaTeX Options

Many aspects of the LaTeX template used to create PDF documents can be customized using top-level YAML metadata (note that these options do not appear underneath the `output` section but rather appear at the top level along with title, author, etc.). For example:

```
---
title: "Crop Analysis Q3 2013"
output: pdf_document
fontsize: 11pt
geometry: margin=1in
---
```

Available metadata variables include:

Variable	Description
lang	Document language code
fontsize	Font size (e.g. 10pt, 11pt, 12pt)
documentclass	LaTeX document class (e.g. article)
classoption	Option for documentclass (e.g. oneside); may be repeated
geometry	Options for geometry class (e.g. margin=1in); may be repeated
mainfont, sansfont, monofont, mathfont	Document fonts (works only with xelatex and lualatex, see the `latex_engine` option)
linkcolor, urlcolor, citecolor	Color for internal, external, and citation links (red, green, magenta, cyan, blue, black)

Advanced Customization

LaTeX Engine

By default PDF documents are rendered using pdflatex. You can specify an alternate engine using the `latex_engine` option. Available engines are “pdflatex”, “xelatex”, and “lualatex”. For example:

```
---
title: "Habits"
output:
  pdf_document:
    latex_engine: xelatex
---
```

Keeping Intermediate TeX

R Markdown documents are converted to PDF by first converting to a TeX file and then calling the LaTeX engine to convert to PDF. By default this TeX file is removed, however if you want to keep it (e.g. for an article submission) you can specify the `keep_tex` option. For example:

```
---
title: "Habits"
output:
  pdf_document:
    keep_tex: true
---
```

Includes

You can do more advanced customization of PDF output by including additional LaTeX directives and/or content or by replacing the core pandoc template entirely. To include content in the document header or before/after the document body you use the `includes` option as follows:

```
---
title: "Habits"
output:
  pdf_document:
    includes:
      in_header: header.tex
      before_body: doc_prefix.tex
      after_body: doc_suffix.tex
---
```

Custom Templates

You can also replace the underlying pandoc template using the `template` option:

```
---
title: "Habits"
output:
  pdf_document:
    template: quarterly_report.tex
---
```

Consult the documentation on pandoc templates (http://johnmacfarlane.net/pandoc/demo/example9/templates.html) for additional details on templates. You can also study the default LaTeX template (https://github.com/jgm/pandoc-templates/blob/master/default.latex) as an example.

Pandoc Arguments

If there are pandoc features you want to use that lack equivilants in the YAML options described above you can still use them by passing custom `pandoc_args` . For example:

```
---
title: "Habits"
output:
  pdf_document:
    pandoc_args: [
      "--no-tex-ligatures"
    ]
---
```

Documentation on all available pandoc arguments can be found in the pandoc user guide (http://johnmacfarlane.net/pandoc/README.html#options).

Shared Options

If you want to specify a set of default options to be shared by multiple documents within a directory you can include a file named `_output.yaml` within the directory. Note that no YAML delimeters or enclosing output object are used in this file. For example:

_output.yaml

```
pdf_document:
  toc: true
  highlight: zenburn
```

All documents located in the same directory as `_output.yaml` will inherit it's options. Options defined explicitly within documents will override those specified in the shared options file.