Number lines in code blocks with Rmarkdown

Atusy

Created on 2019-04-18.  
Updated on 2019-04-19.

**Before reading**

* For quick start, go “[Autonumbering](#autonumbering)”.
* A source and output examples in variety of formats are on GitHub [atusy/rmd-line-num](https://github.com/atusy/rmd-line-num).
* Ask me questions on [GitHub Issues](https://github.com/atusy/rmd-line-num/issues) or on Twitter ([@Atsushi776](https://twitter.com/Atsushi776))

# Number Pandoc’s fenced code

Pandoc has an official support to number lines on fenced code by giving a class attribute numberLines (<https://www.pandoc.org/MANUAL.html#extension-fenced_code_attributes>). The document also introduces that lineAnchors class makes line numbers be clickable, and startFrom arugment supports specify starting number.

Note that class attributes require . before the class name.

```{.r .numberLines .lineAnchors startFrom="11"}  
x <- rnorm(10)  
mean(x)  
```

becomes

x <- rnorm(10)  
mean(x)

This is enough for rmarkdown::pdf\_document.

For html\_document, we also need to specify highlight other than default in YAML front matter[[1]](#footnote-25), [[2]](#footnote-27). See ?rmarkdown::html\_document for available highlight options.

---  
title: Line numbers with Rmarkdown documents  
output:   
 html\_document:  
 highlight: pygment  
 pdf\_document: default  
---

# Number Rmarkdown’s chunks and outputs

The above success infers a success in chunks of Rmarkdown.

## Numbering chunks

For code chunks of Rmarkdown documents, numberLines class can be given by assigning class.source = "numberLines" as a chunk option[[3]](#footnote-32). You may also add lineAnchors demilited by space ("numberLines lineAnchors") or as another element of vector (c("numberLines", "lineAnchors")) [[4]](#footnote-34). Unfortunately, I find no way to add startFrom="integer".

Again, you need to specify highlight for html\_document.

Thus,

---  
title: Line numbers with Rmarkdown documents  
output:   
 html\_document:  
 highlight: pygment  
---  
  
```{r, class.source = "numberLines lineAnchors"}  
x <- seq(10)  
mean(x)  
```

becomes

x <- seq(10)  
mean(x)

## [1] 5.5

Great again!!

Unfortunately, line numbering does not work on rmarkdown::word\_document.

It does not work properly on rmarkdown::html\_notebook and blogdown::html\_page as well. I guess some tricks required in CSS or JS.

## Numbering outputs

You may also want to number lines on outputs by class.output = "numberLines" with or without lineAnchors just like input. However, this changes background colors to gray.

## [1] 5.5

If output format is html, css will help.

Before adding class.output = "numberLines", an output in html is

<pre><code>## [1] 5.5</code></pre>

After adding it, the output in html becomes

<div class="sourceCode" id="cb7"><pre class="sourceCode numberSource numberLines"><code class="sourceCode"><a class="sourceLine" id="cb7-1" title="1">## [1] 5.5</a></code></pre></div>

When chunkout class is further added, the output in html becomes

<div class="sourceCode" id="cb10"><pre class="sourceCode numberSource numberLines chunkout"><code class="sourceCode"><a class="sourceLine" id="cb10-1" title="1">## [1] 5.5</a></code></pre></div>

You can see chunkout class is added to pre tag.

So, lets modify css with

div.sourceCode pre.chunkout {  
 background: white;  
}

and you’ll be happy, right?

## [1] 5.5

Sorry I do not support pdf because kills me.

# Autonumbering

* Super easy by a following template and edit after <!-- Start your body -->.
* Disable numbering by class.source = NULL, class.output = NULL.
* Note autonumbering does not work on Pandoc’s fenced code [[5]](#footnote-39).
* ---  
  output:  
   html\_document:  
   highlight: pygment  
   pandoc\_args:  
   - --indented-code-classes  
   - lineNumbers  
  ---

## Input with a template

---  
output:  
 html\_document:  
 highlight: pygment  
 pdf\_document: default  
---  
  
```{r setup, include = FALSE}  
knitr::opts\_chunk$set(  
 class.source = "numberLines lineAnchors",   
 class.output = c("numberLines lineAnchors chunkout")   
)  
  
# Add some arbitrary setup codes  
```  
  
  
```{css, echo = FALSE}  
div.sourceCode pre.chunkout {  
 background: white;  
}  
```  
  
<!-- Start your body -->  
  
\*\*Numbered\*\*  
  
```{r}  
x <- seq(10)  
mean(x)  
```  
  
\*\*Unnumbered\*\*  
  
```{r, class.source = NULL, class.output = NULL}  
x <- seq(10)  
mean(x)  
```

## Output from the template

**Numbered**

x <- seq(10)  
mean(x)

## [1] 5.5

**Unnumbered**

x <- seq(10)  
mean(x)

## [1] 5.5

# Enjoy !

1. “Chunk numberLines hook]” on Rpubs figured out requirement of pygment (<https://rpubs.com/Thell/numberLines>). [↑](#footnote-ref-25)
2. @niszet0 mentioned that not only pygment but also highlight other than default are acceptable (<https://niszet.hatenablog.com/entry/2019/04/18/194246> in Japanese). [↑](#footnote-ref-27)
3. See “Tex Results” section of “Chunk options and package options” by Yi Hui (<https://yihui.name/knitr/options/#text-results>) [↑](#footnote-ref-32)
4. [@niszet](https://twitter.com/niszet0) mentioned me on Twitter <https://twitter.com/niszet0/status/1118837863200591872> [↑](#footnote-ref-34)
5. The document indicates a following YAML front matter should work, but doesn’t (<https://www.pandoc.org/MANUAL.html#reader-options>). [↑](#footnote-ref-39)