PDFs and PNGs

If you want to save your graphs as PDFs, then simply set

```
knitr::opts_chunk$set(dev = "cairo_pdf")
```

at the top of Rmarkdown file. The PNG variant is slightly different as we need to specify the device `dev` and also pass the `type` argument to the device

```
knitr::opts_chunk$set(dev = "png", dev.args = list(type = "cairo-png"))
```

These options, i.e. `dev = "cairo_pdf"`, can also be set at individual chunks.

The ragg Package

Setting the `agg_png()` function from the **ragg** package as the graphics device is somewhat more tricky as it doesn't come pre-defined within **knitr**. The **knitr** docs states that

> if none of the 20 built-in devices is appropriate, we can still provide yet another name as long as it is a legal function name which can record plots (it must be of the form function(filename, width, height))

The arguments of `agg_png()` are

```
formals(ragg::agg_png)[1:3]
#> $filename
#> [1] "Rplot%03d.png"
#>
#> $width
#> [1] 480
#>
#> $height
#> [1] 480
```

This suggests we can simply set `ragg::agg_png()` as the **knitr** `dev`, as it's of the correct form. However, careful reading of the knitr source code highlights that the `dpi` argument isn't passed to new devices and that the units should be inches. So after a "little" experimentation, we have

```
ragg_png = function(..., res = 192) {
  ragg::agg_png(..., res = res, units = "in")
}
knitr::opts_chunk$set(dev = "ragg_png", fig.ext = "png")
```

Remember the `dpi` argument isn't passed to `ragg_png()`, so if you want to change the resolution per chunk, then you will need to use

```
dev.args = list(ragg_png = list(res = 192))
```

As **ragg** is being developed by RStudio, I'm guessing that at some point in the near future, ragg will become native to **knitr**.