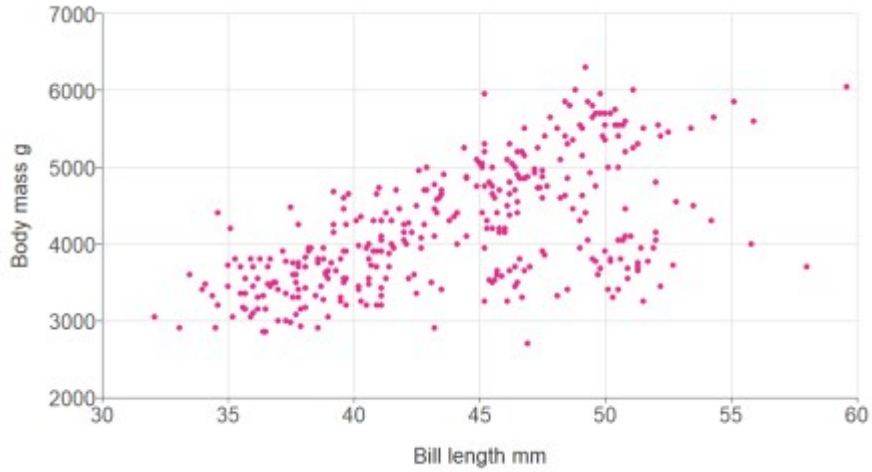
1. Always define the colours to use via the `pal` argument

The colour palette can be changed from the default viridis colours by providing a character vector of hex codes to the `pal` argument.

```
gg_point(penguins, bill_length_mm, body_mass_g, pal = "#e7298a")
```

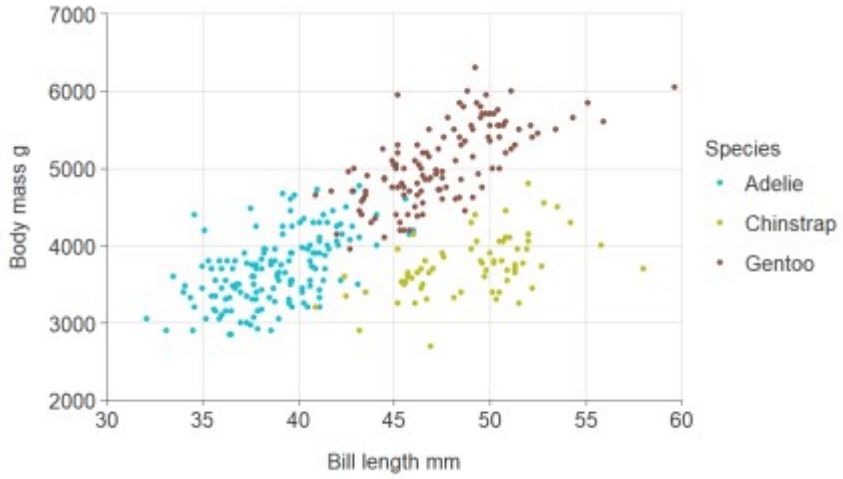


Users can get access to a large amount of colour palettes through the `pals` package.

2. If colouring by a variable, use a `*_col()` or `*_col_facet()` function, and define the `col_var`

To colour by a variable, use a `*_col()` function and then define that variable to be coloured using the `col_var` argument.

```
gg_point_col(penguins, bill_length_mm, body_mass_g, species)
```



3. For `gg_sf_col*()` and `gg_point_col*()` functions where colouring by a numeric variable, also define the `col_method` and `col_cuts`

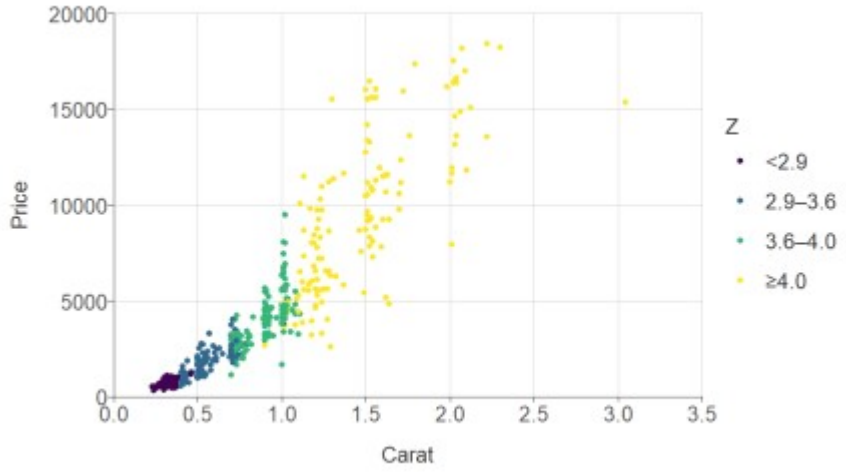
All `simplevis` `*_col()` and `*_col_facet()` functions support colouring by a categorical variable.

In addition, `sf` and `point *_col()` and `*_col_facet()` functions support colouring by a numeric variable.

You do this by specifying whether you want to do this by:

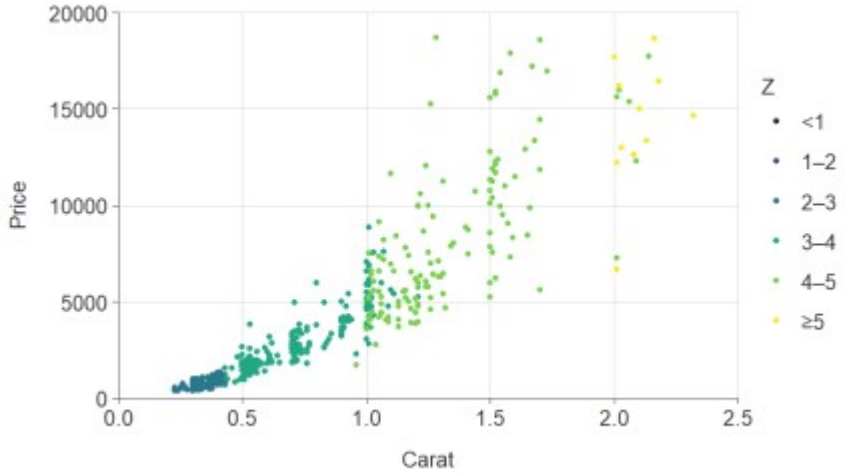
- defining whether the col_method is to be by `bin` or `quantile`
- defining a vector or col_cuts. These should be between 0 and infinity (`Inf`) for `bin` and between 0 and 1 for `quantile`

```
plot_data <- ggplot2::diamonds %>%
  slice_sample(prop = 0.01)

plot_data
#> # A tibble: 539 x 10
#>    carat cut       color clarity depth table price     x     y     z
#>    <dbl> <ord>     <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>
#>  1  1.22 Ideal     E     VS1      62.4    54 10622  6.77  6.88  4.26
#>  2  0.56 Premium   D     SI1      61      60  1605  5.28  5.25  3.21
#>  3  0.33 Very Good E     VS1      58.3    62   886  4.49  4.57  2.64
#>  4  0.52 Premium   D     SI1      61.5    55  1651  5.21  5.19  3.2
#>  5  1.02 Very Good F     VS1      62.8    58  7539  6.4   6.44  4.03
#>  6  1.52 Good      F     VS2      64.2    59 11696  7.16  7.2   4.61
#>  7  0.37 Ideal     F     VS1      61.1    56   846  4.64  4.65  2.84
#>  8  0.83 Ideal     G     VS1      62.1    55  4989  6.02  6.05  3.75
#>  9  0.32 Ideal     F     VS1      61.6    57   716  4.39  4.42  2.71
#> 10  1.06 Premium   D     SI2      61.6    61  4903  6.59  6.53  4.04
#> # ... with 529 more rows

gg_point_col(plot_data,
                 x_var = carat,
                 y_var = price,
                 col_var = z,
                 col_method = "quantile",
                 col_cuts = c(0, 0.25, 0.5, 0.75, 1))
```




```
gg_point_col(plot_data,
                 x_var = carat,
                 y_var = price,
```

```
                col_var = z,
                col_method = "bin",
                col_cuts = c(0, 1, 2, 3, 4, 5, Inf))
```



Further information

More blogs to come on `simplevis`. In the meantime, see the vignette and articles on the simplevis website.