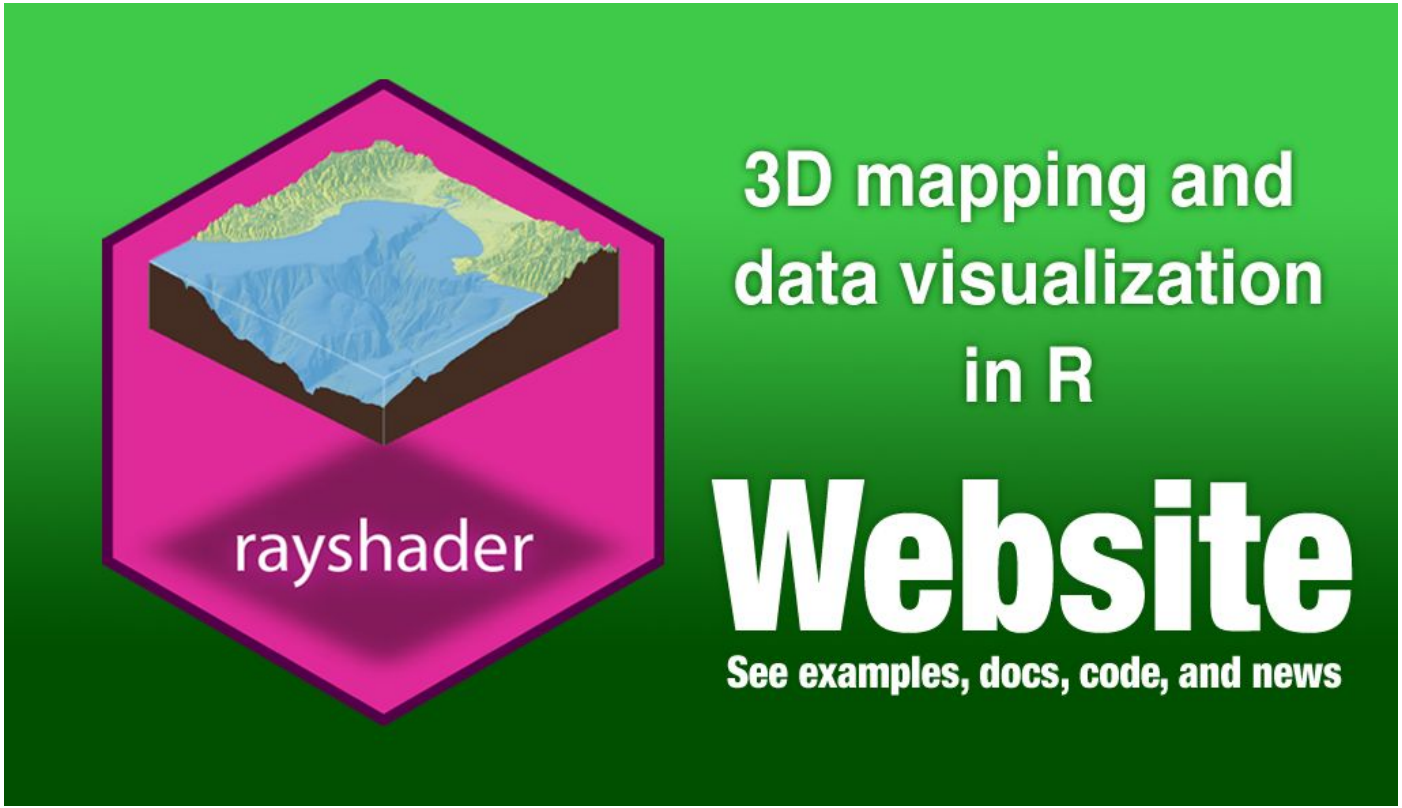
rayshader

W krainie R freaków
oraz
Jak zrobić najgorszy wykres w karierze

Z czym będziemy się mierzyć?



Plan prezentacji

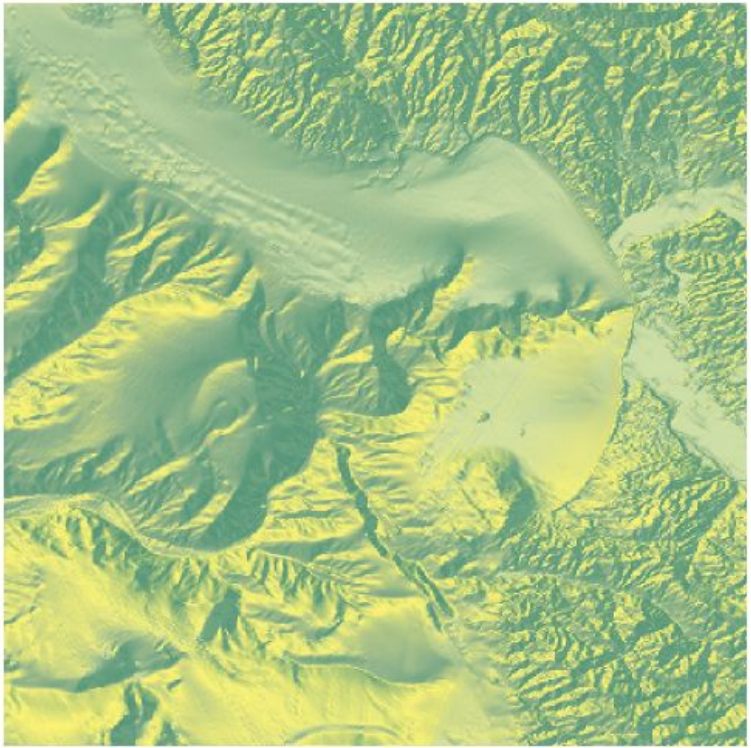
- Czym jest generowanie powierzchni?
- Od numerycznego modelu terenu do wizualizacji z rayshaderem
- Funkcje poboczne pakietu
- Inne funkcje poboczne
- Jeszcze inne funkcje poboczne aka Ostateczny Przewodnik Po Robieniu Mega Dziwnych Wykresów

Czym jest generowanie powierzchni?

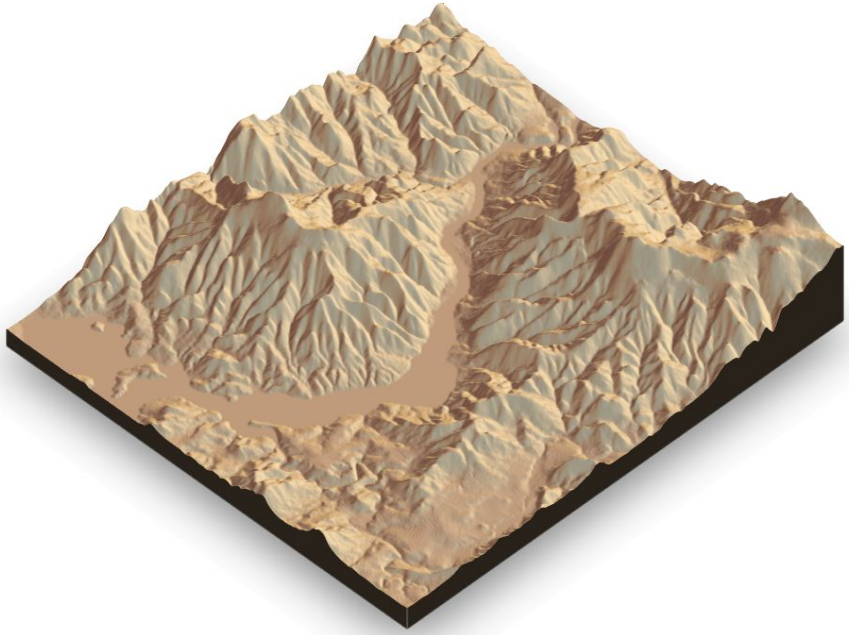
Źródło: https://epodreczniki.pl/a/jaki-ksztalt-moze-miec-powierzchnia-ziemi/D9We4dIYp

Od numerycznego modelu terenu do wizualizacji z rayshaderem

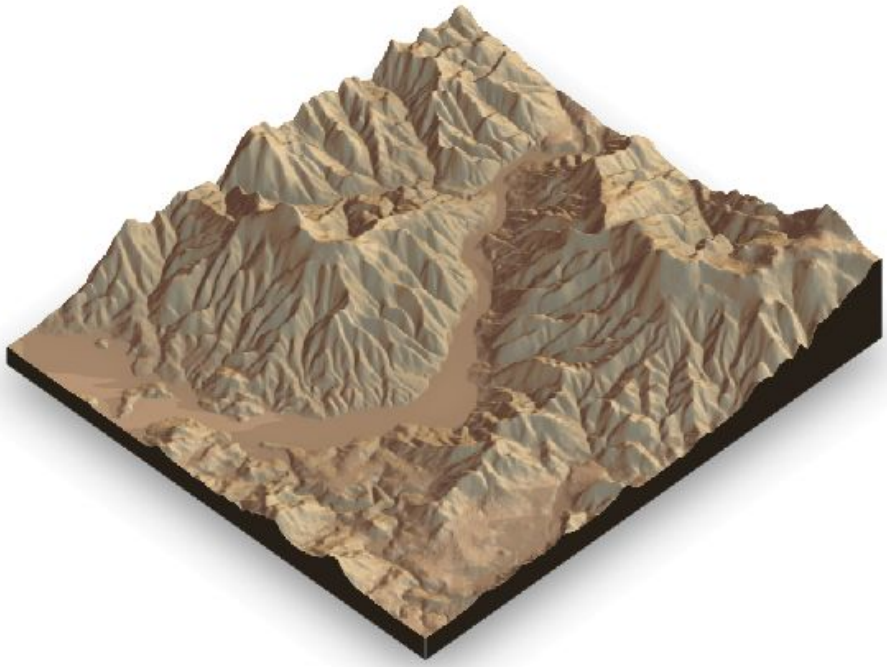
```
plot_map(sphere_shade(montereybay))
```



```
data %>%
  sphere_shade(texture = "desert") %>%
  #add_shadow(ray_shade(elevation_matrix)) %>%
  #add_shadow(ambient_shade(elevation_matrix)) %>%
  plot_3d(data, zscale = 10, fov = 0, theta = 135, zoom = 0.75, phi = 45, windowsize = c(1000, 800))
render_snapshot()
```

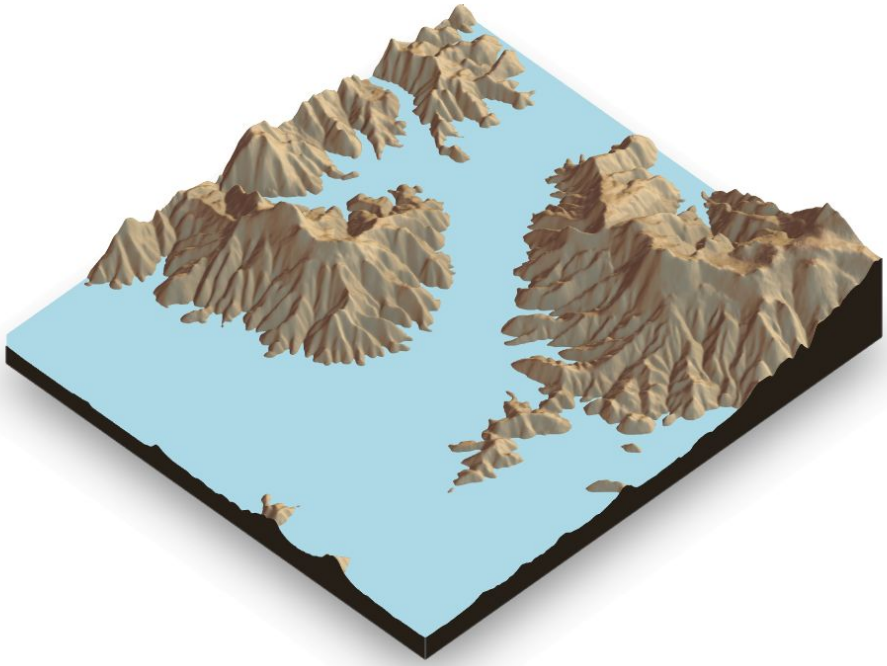


```
data %>%
  sphere_shade(texture = "desert") %>%
  add_shadow(ray_shade(elevation_matrix)) %>%
  add_shadow(ambient_shade(elevation_matrix)) %>%
  plot_3d(data, zscale = 10, fov = 0, theta = 135, zoom = 0.75, phi = 45, windowsize = c(1000, 800))
render_snapshot()
```



```
data %>%
  sphere_shade(texture = "desert") %>%
  add_water(detect_water(data), color = "desert") %>%
  add_shadow(ray_shade(data, zscale = 3), 0.5) %>%
  add_shadow(ambient_shade(data), 0.5) %>%
  plot_3d(data, zscale = 10, fov = 0, theta = 135, zoom = 0.75, phi = 45, windowsize = c(1000, 800))
```

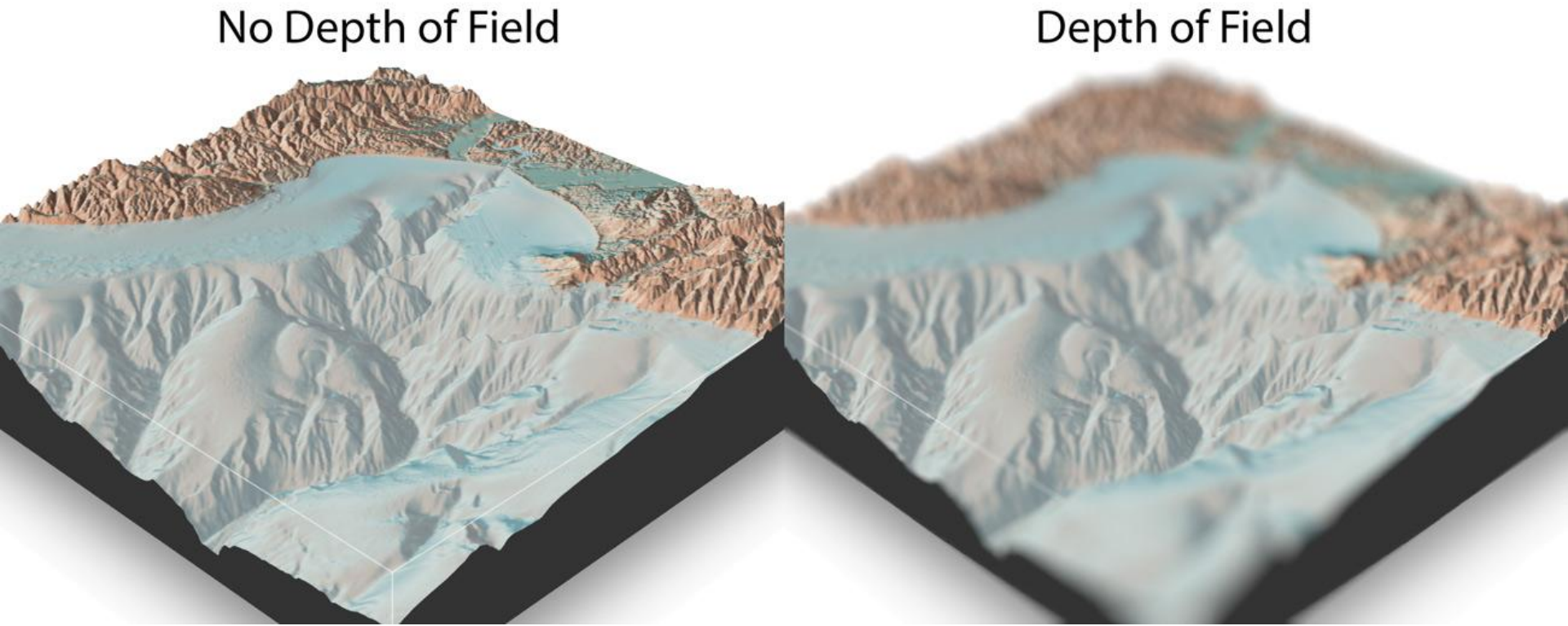
```
data %>%
  sphere_shade(texture = "desert") %>%
  add_shadow(ray_shade(data)) %>%
  add_shadow(ambient_shade(data)) %>%
  plot_3d(data,zscale = 10, theta = 135,fov=0,waterdepth = 150, zoom = 0.75, phi = 45, water = TRUE,wateralpha = 1, windowsi
ze = c(1000, 800))
```



Funkcje poboczne pakietu

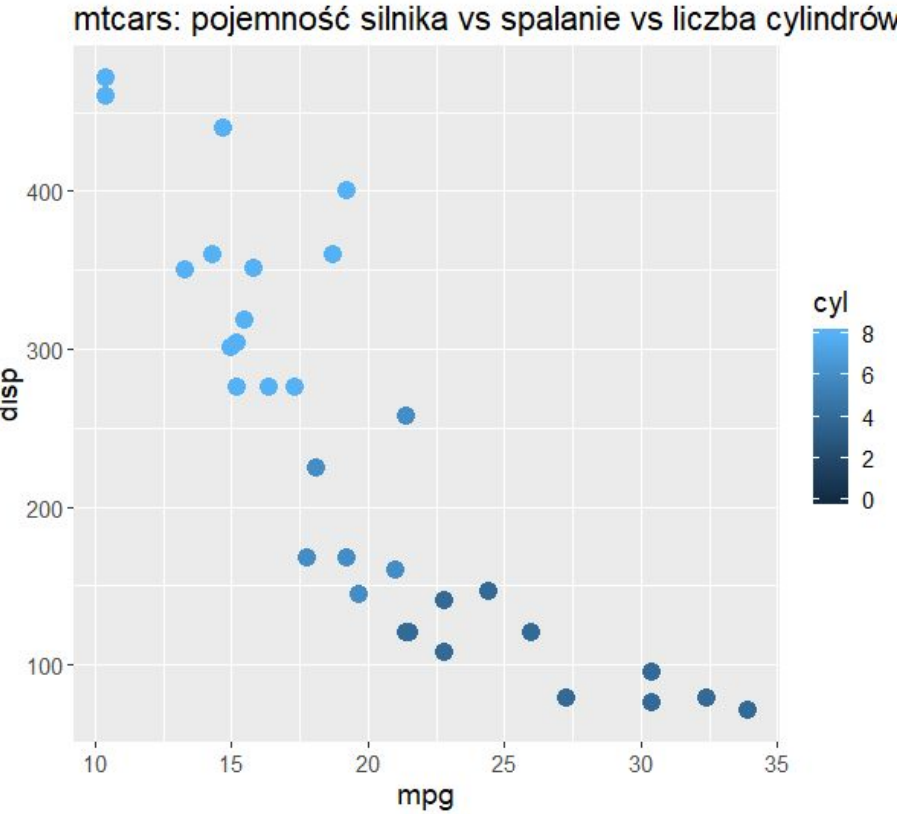
- save_3dprint(...)
- save_obj(...)
- save_png(...)
- render_camera(...)
- render_highquality(...)
- render_movie(...)

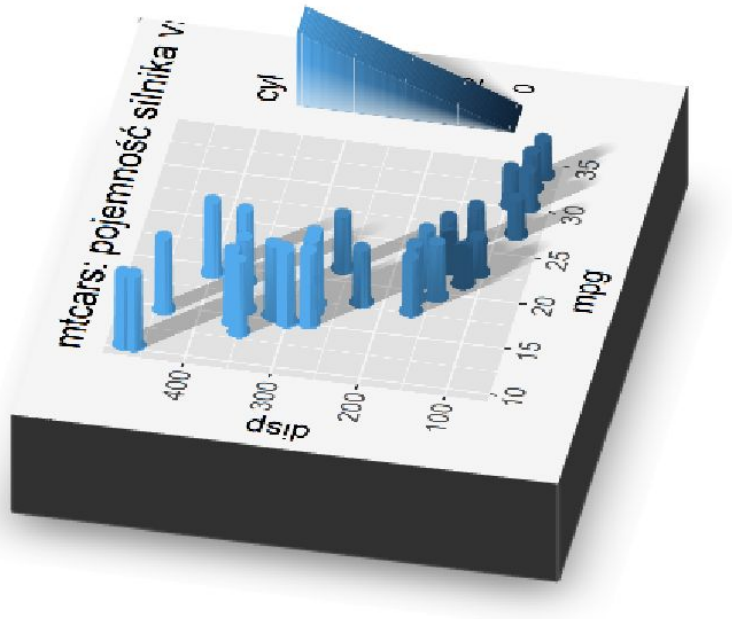
Inne funkcje poboczne - głębia ostrości



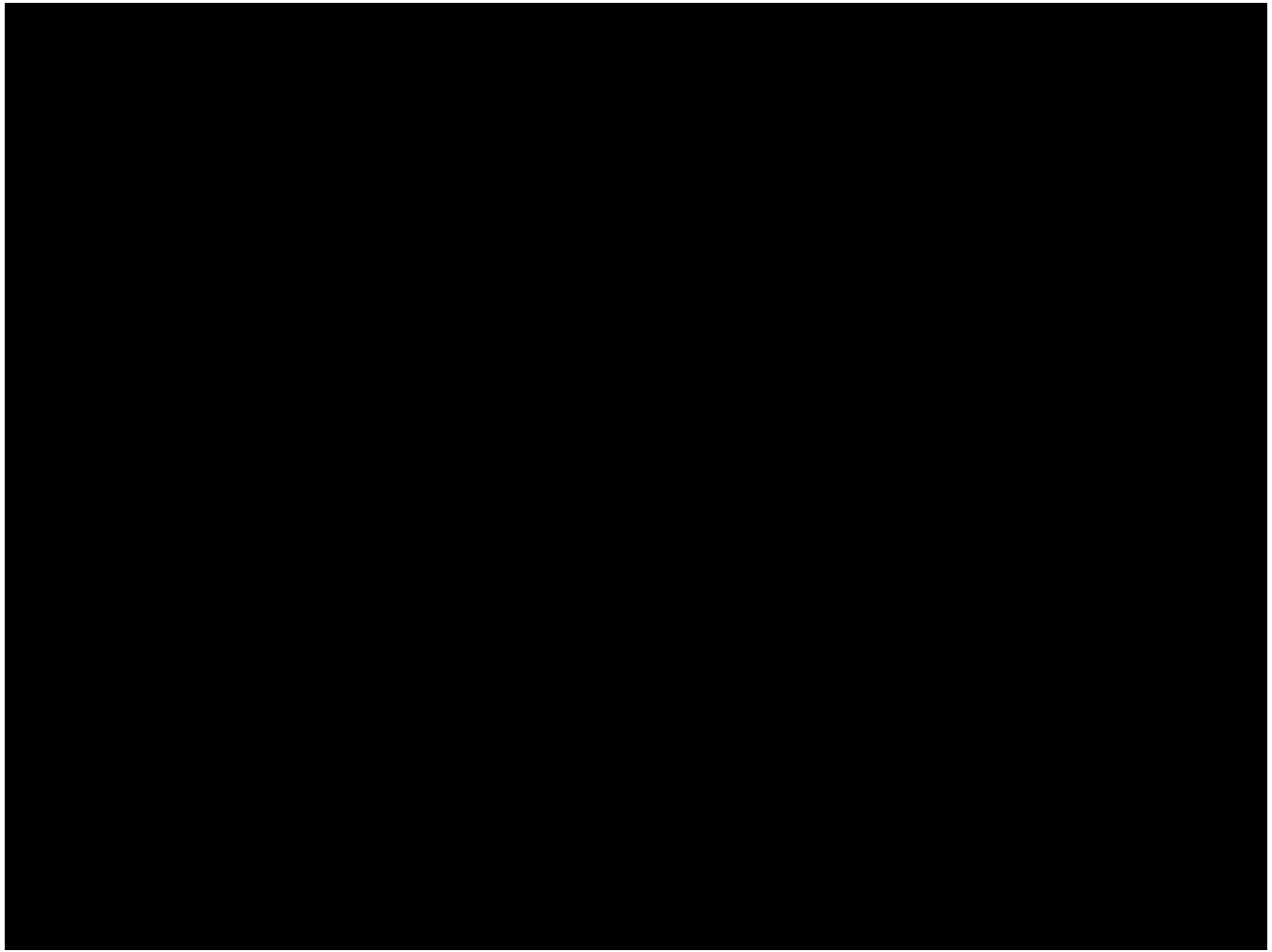
Inne funkcje poboczne - głębia ostrości

Jeszcze inne funkcje poboczne

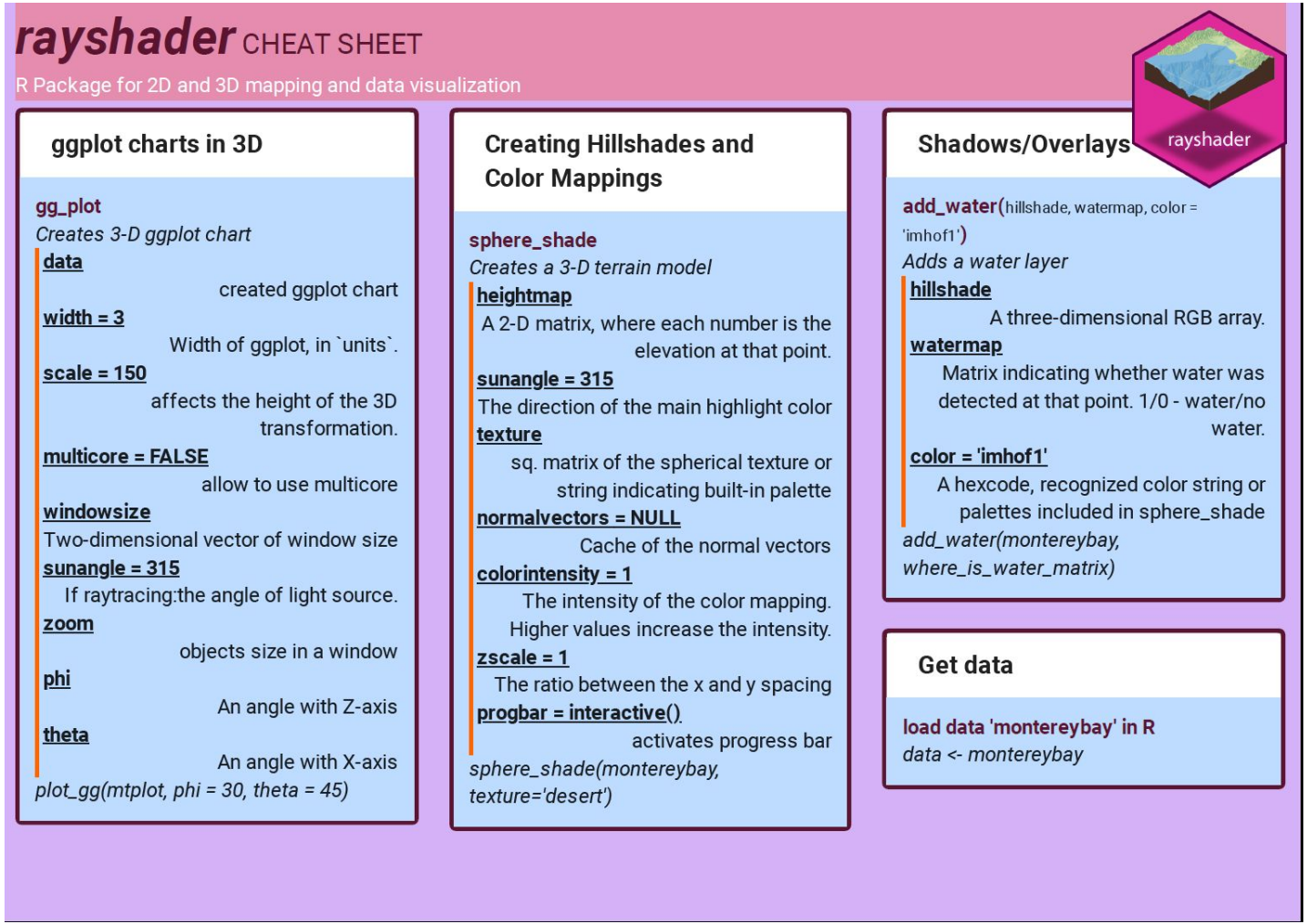




???



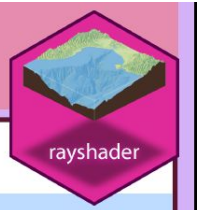
Cheatsheet


rayshader CHEAT SHEET
R Package for 2D and 3D mapping and data visualization
rayshader
ggplot charts in 3D
gg_plot
Creates 3-D ggplot chart
data
created ggplot chart
width = 3
Width of ggplot, in `units`.
scale = 150
affects the height of the 3D transformation.
multicore = FALSE
allow to use multicore
windowsize
Two-dimensional vector of window size
sunangle = 315
If raytracing:the angle of light source.
zoom
objects size in a window
phi
An angle with Z-axis
theta
An angle with X-axis
plot_gg(mtplot, phi = 30, theta = 45)
Creating Hillshades and Color Mappings
sphere_shade
Creates a 3-D terrain model
heightmap
A 2-D matrix, where each number is the elevation at that point.
sunangle = 315
The direction of the main highlight color
texture
sq. matrix of the spherical texture or string indicating built-in palette
normalvectors = NULL
Cache of the normal vectors
colorintensity = 1
The intensity of the color mapping. Higher values increase the intensity.
zscale = 1
The ratio between the x and y spacing
progbar = interactive()
activates progress bar
sphere_shade(montereybay, texture='desert')
Shadows/Overlays
add_water(hillshade, watermap, color = 'imhof1')
Adds a water layer
hillshade
A three-dimensional RGB array.
watermap
Matrix indicating whether water was detected at that point. 1/0 - water/no water.
color = 'imhof1'
A hexcode, recognized color string or palettes included in sphere_shade
add_water(montereybay, where_is_water_matrix)
Get data
load data 'montereybay' in R
data <- montereybay

Cheatsheet

rayshader CHEAT SHEET

R Package for 2D and 3D mapping and data visualization



Plotting and Saving 2D and 3D Visualizations

plot_3d(...)
Plot 3D
plot_map(...)
Plot Map
save_3dprint(...)
Save 3D Print
save_obj(...)
Save OBJ
save_png(...)
Save PNG

Adding Shadows and Overlays to Maps

add_overlay(...)
Add Overlay
add_shadow(...)
Add Shadow
detect_water(...)
Detect water

Creating Hillshades and Color Mappings

ambient_shade(...)
Calculate Ambient Occlusion Map
height_shade(...)
Calculate Terrain Color Map
lamb_shade(...)
Calculate Lambert Shading Map
ray_shade(...)
Calculate Raytraced Shadow Map
calculate_normal(...)
Calculate Normal
create_texture(...)
Create Texture

Capture 3D Maps

render_camera(...)
Render Camera
render_depth(...)
Render Depth of Field
render_highquality(...)
Render High Quality
render_label(...)
Render Label
render_movie(...)
Render Movie
render_water(...)
Render Water Laye