

## 5.1 How to handle Raster Data

# Raster Data in QGIS

## After this lesson you can ...

- Load in vector data in QGIS and R
- How to visualise vector data

# Load in your own Raster Data

1. Download the raster data
2. Move the folder of the raster to your desired location
3. Load in the Data

## Method 1)

Just pull the raster file from  
your explorer into QGIS

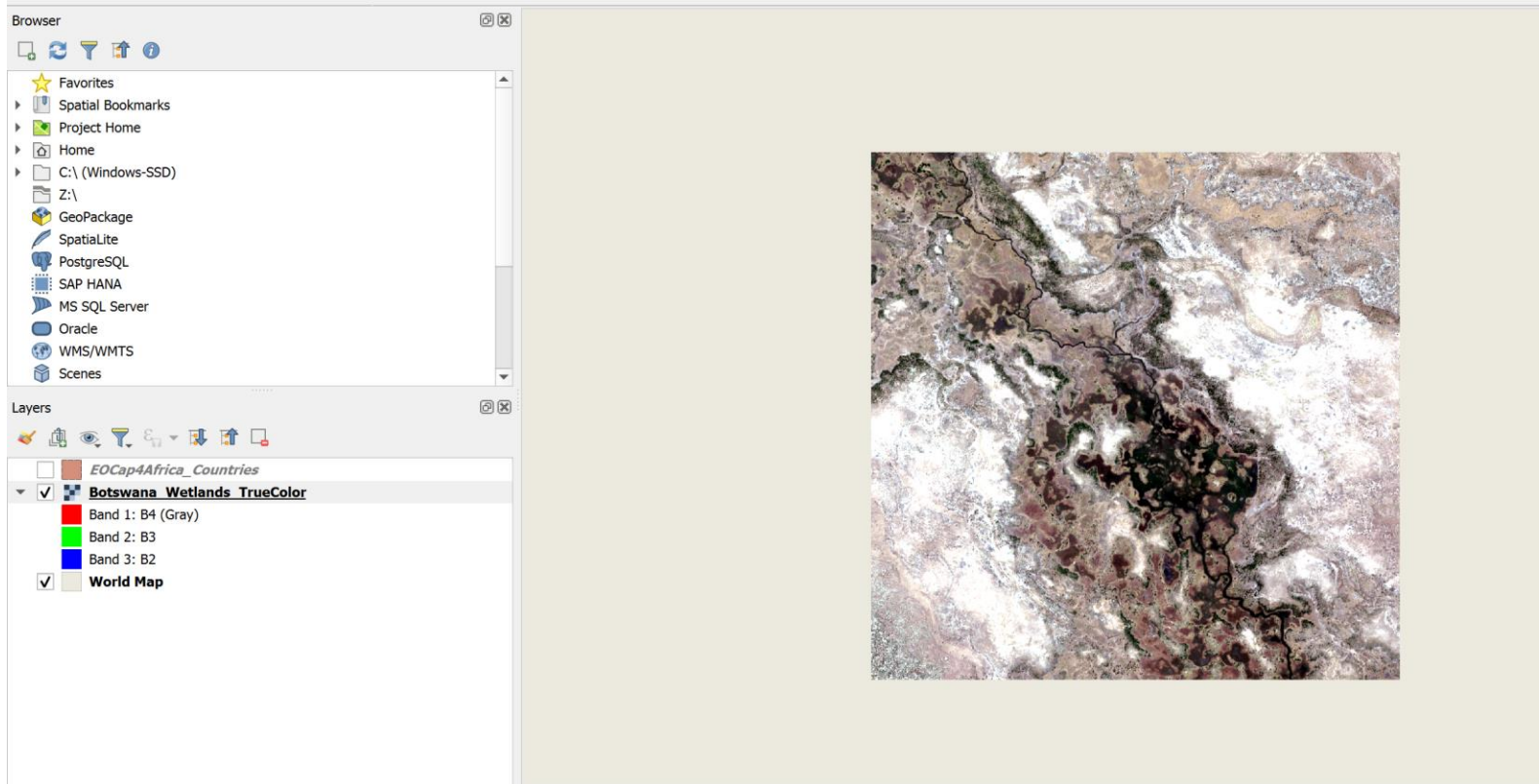
## Method 2)

- 1) Go to Layer in the Menu Bar
- 2) Add Layer
- 3) Add Raster Layer
- 4) Go to the raster and  
doubleclick it

# Inspecting Raster Data

Tip!

Right click on your layer and select "Zoom to layer" to immediately switch the map view to the layer's extent



Inspect the properties of the raster file. Are they different compared to the vector file?

# Inspecting the bands

Switch out the bands in the layer properties and create false-color images

Can you think of cases where you would want to switch the bands?

Layer Properties - Botswana\_Wetlands\_TrueColor — Symbology

**Band Rendering**

Render type: Multiband color

Red band: Band 1: B4 (Gray) - B4  
Min: 1456 Max: 3257,5

Green band: Band 2: B3 - B3  
Min: 1531 Max: 2770

Blue band: Band 3: B2 - B2  
Min: 1367,5 Max: 2446,99

Contrast enhancement: Stretch to MinMax

**Min / Max Value Settings**

**Layer Rendering**

Blending mode: Normal

Brightness: 0 Contrast: 0

Gamma: 1,00 Saturation: 0

☐ Invert colors Grayscale: Off

Hue: ☐ Colorize Strength: 100%

**Resampling**

Zoomed: in Nearest Neighbour out Nearest Neighbour Oversampling: 2,00 ☐ Early resampling

Style OK Cancel Apply Help

# Vector Data in R

# Start Rstudio

1. Double click on the downloaded and unzipped R-file  
"5.2\_How\_to\_handle\_Raster\_Data.r"
2. The file will open in RStudio



# Using packages

```
# 1. Install and load necessary packages  
install.packages(c("terra", "ggplot2"))  
  
# Load the libraries  
library(terra)      # For handling raster data  
library(ggplot2)    # For visualization
```

Do you still remember what the "c" does?

# Get the Raster Data

- 1) We are using the raster we already visualised in QGIS
- 2) Define the variable "raster\_path" using the file path where you placed the .tif file

```
# 2. Load raster data  
# Replace this path with the path to your raster file  
raster_path <- "C:/Users/schu1/Documents/Berufliches/Hiwi EOCap4Africa/Geodata_tasks/  
  
# Read the raster file into R using `rast` from `terra`  
true_color <- rast(raster_path)
```

- 3) We assign the raster to the variable "elevation"

# Assess the metadata

- Print() gives you the most basic information about your raster
- Summary() displays statistics as well as a range of information about the geometry

```
# Check the structure of the raster data  
print(true_color) # Displays basic information about the raster  
summary(true_color) # Provides statistical summaries of pixel values
```

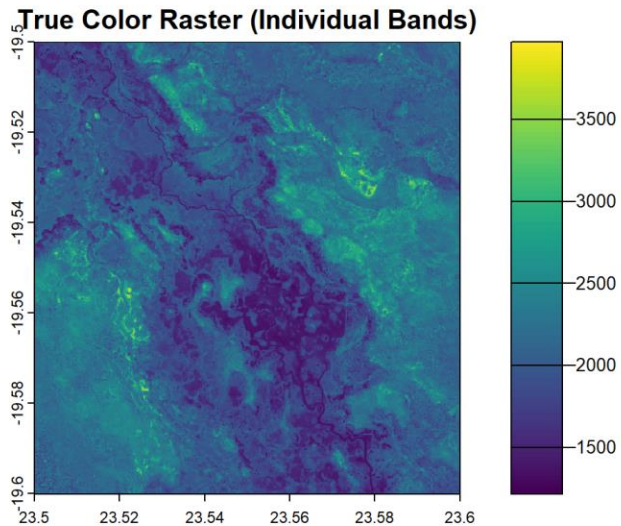
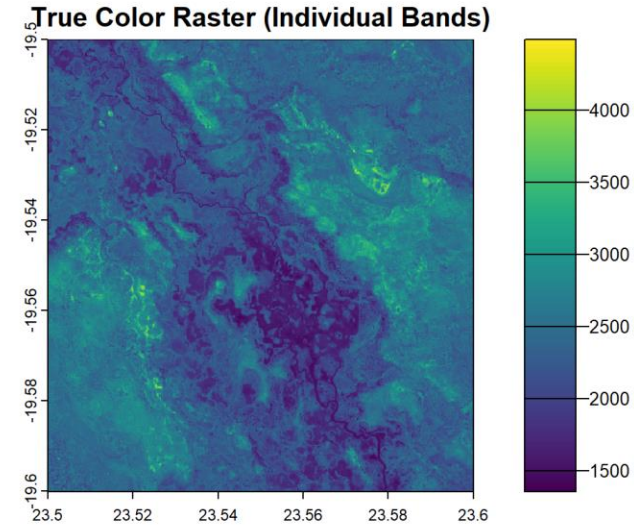
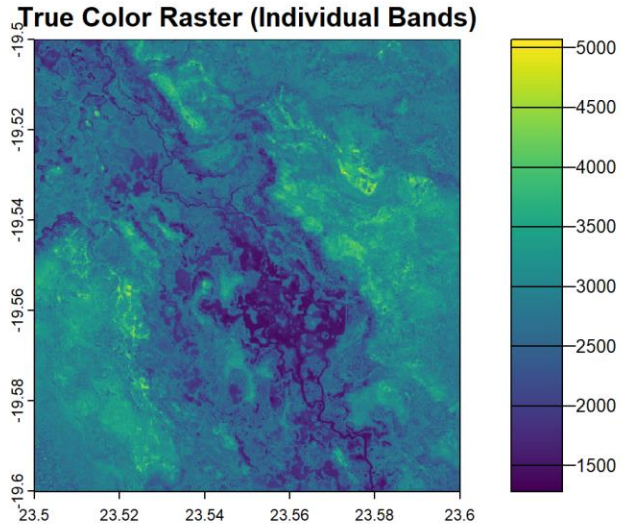
# Visualise Raster Data

- Use the basic plot() function to display any type of data
- Using "main" we define the plot title

```
# 3. Visualize the raster data  
# Quick visualization of individual bands  
plot(true_color, main = "True Color Raster (Individual Bands)")
```

# Visualise Raster Data

The result:



So, why are there 3 images?

Because the raster data has 3 bands!

# Visualise Raster Data

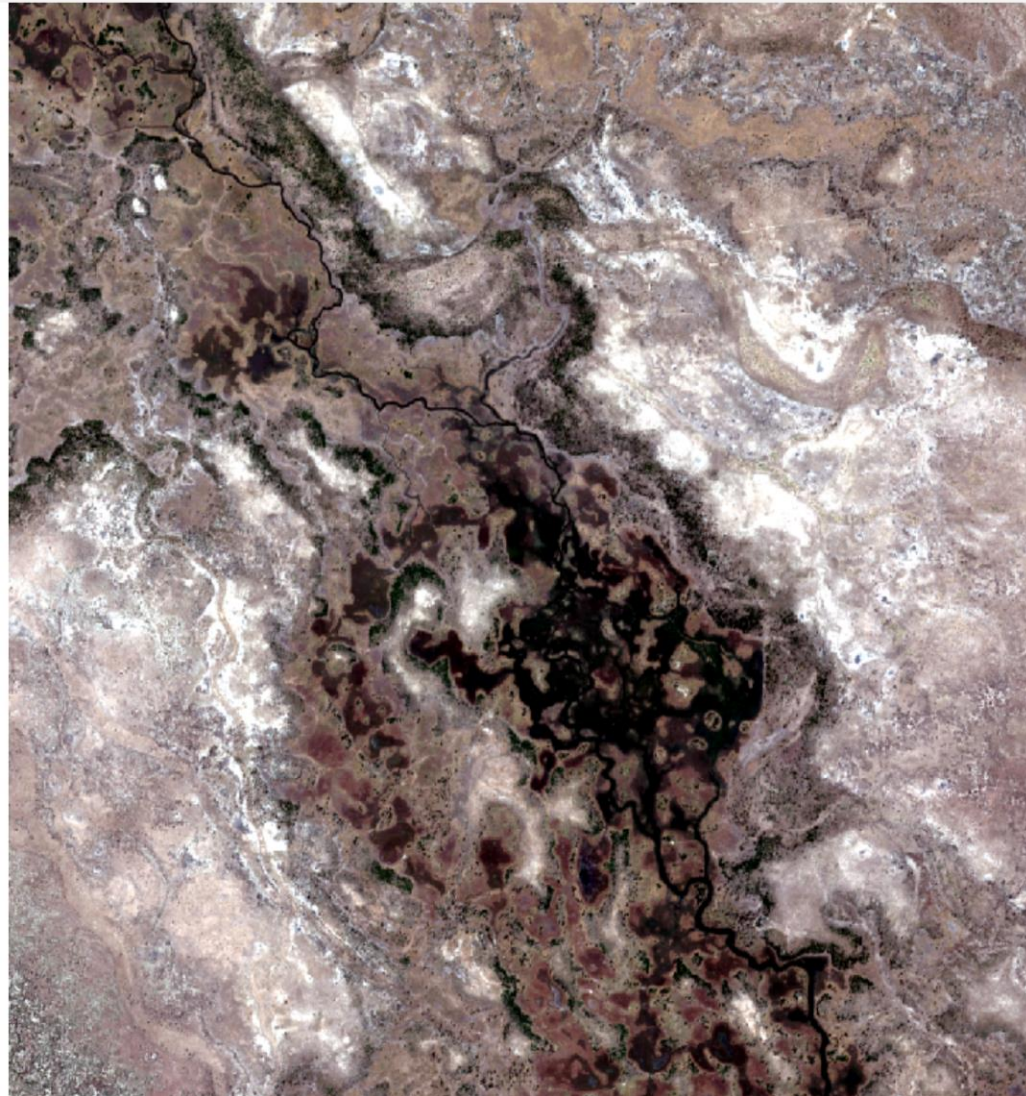
- To visualise Raster Data as a true color image we need to assign the rasters bands to the variables "r" (red), "g" (green), "b" (blue)

```
# Combine bands for visualization as a true-color image  
# terra::plotRGB() is used for RGB visualization  
plotRGB(true_color, r = 1, g = 2, b = 3, stretch = "lin",  
        main = "True Color Image")
```



# Visualise Raster Data

The result:



# Visualise Raster Data

## Your turn

- Create a false color image in R