1. Create Maps With `R` Geospatial Classes and Graphics Tools (*Making Maps*)
2. Read and write ESRI Shape Files (*ESRI*)
3. Display T Spatial Objects with Google Maps and Google Earth (*KML*)
4. Read and Display Data from GPS Devices Using `R` (*GPX*)
5. Overlay Points on Satellite Image / Extract Pixel Values (*Raster*)

ESRI Shapefiles with R

ESRI shapefiles

- The ESRI Shapefile is a widely used file format for storing vector-based geopatial data (i.e., points, lines, and polygons).
- This example demonstrates use of several different `R` packages that provide functions for reading and/or writing shapefiles.

ESRI shapefiles

Demonstration

- The first two approaches shown here use packages (specifically **rgdal** and **maptools**) that depend on the **sp** package, which defines a set of spatial classes that have become the de facto standard spatial data types in R.
- The third approach (PBS Mapping) creates R data objects that are less generally useful, but necessary when calling other analytical functions defined in the package.

Using rgdal

- The rgdal package provides an interface to the **GDAL/OGR** library, which powers the data import/export capabilities of many geospatially aware software applications.
- The package includes functions `readOGR` and `writeOGR` for reading and writing not only shapefiles, but numerous other vector-based file formats.
- In addition, the `ogrInfo` function is useful for retrieving details about the file without reading in the full dataset. These functions are all capable of automatically reading and writing projection information if available.

Using rgdal

- Provided you are able to install the separate **GDAL/OGR** library - which may be tricky on some systems - it is worth learning how to use this package if you frequently work with shapefiles and/or other spatial data formats, including not just vector formats but raster formats as well.

Using maptools:

- The **maptools** package includes a number of useful functions for reading, writing, converting, and otherwise handling spatial objects in *R*.
- The general functions for reading and writing shapefiles are `readShapeSpatial` and `writeSpatialShape`, respectively.
- In both cases, the function automatically determines whether the shapefile (or `R` object) contains points, lines, or polygons, and will then read in (or write out) the data using a more specialized function of the particular type.

Using maptools

- These specialized functions, such as `readShapeLines` for reading lines, can also be called directly.
- One advantage of doing so is that it will complain if you inadvertently use it on the wrong data type, helping you to catch errors sooner.
- Unlike their **rgdal** counterparts, the ma**ptools** functions neither read nor write projection information, leaving it up to you to manage these details manually.

About PBSMapping (CRAN Description)

PBSmapping:
Mapping Fisheries Data and Spatial Analysis Tools:
This software has evolved from fisheries research conducted at the **Pacific Biological Station** (PBS) in Nanaimo, British Columbia, Canada. It extends the R language to include two-dimensional plotting features similar to those commonly available in a Geographic Information System (GIS).
Embedded C code speeds algorithms from computational geometry, such as finding polygons that contain specified point events or converting between longitude-latitude and Universal Transverse Mercator (UTM) coordinates.

Using PBSmapping

- The PBSmapping package can also read (but not write) shapefiles.
- However, note that **PBSmapping** uses its own custom-defined spatial data types that are optimized to work with various specialized package functions.
- This makes it harder to take advantage of functions defined in the numerous packages that are built on **sp**, although the **maptools** package does provide functions that convert between the different formats.