Statistical Modelling II

HES 505 Fall 2024: Session 22

Carolyn Koehn

Objectives

By the end of today you should be able to:

- Articulate the differences between statistical learning classifiers and logistic regression
- Describe several classification trees and their relationship to Random Forests
- Describe MaxEnt models for presence-only data

Revisiting Classification

Favorability in General

- Logistic regression treats as a (generalized) linear function
- Allows for multiple qualitative classes
- Ensures that estimates of are [0,1]

Key assumptions of logistic regression

- Dependent variable must be binary
- Observations must be independent (important for spatial analyses)
- Predictors should not be collinear
- Predictors should be linearly related to the log-odds
- **Sample Size**

Beyond Linearity

- Logistic (and other generalized linear models) are relatively interpretable
- Probability theory allows robust inference of effects
- Predictive power can be low
- Relaxing the linearity assumption can help

Classification Trees

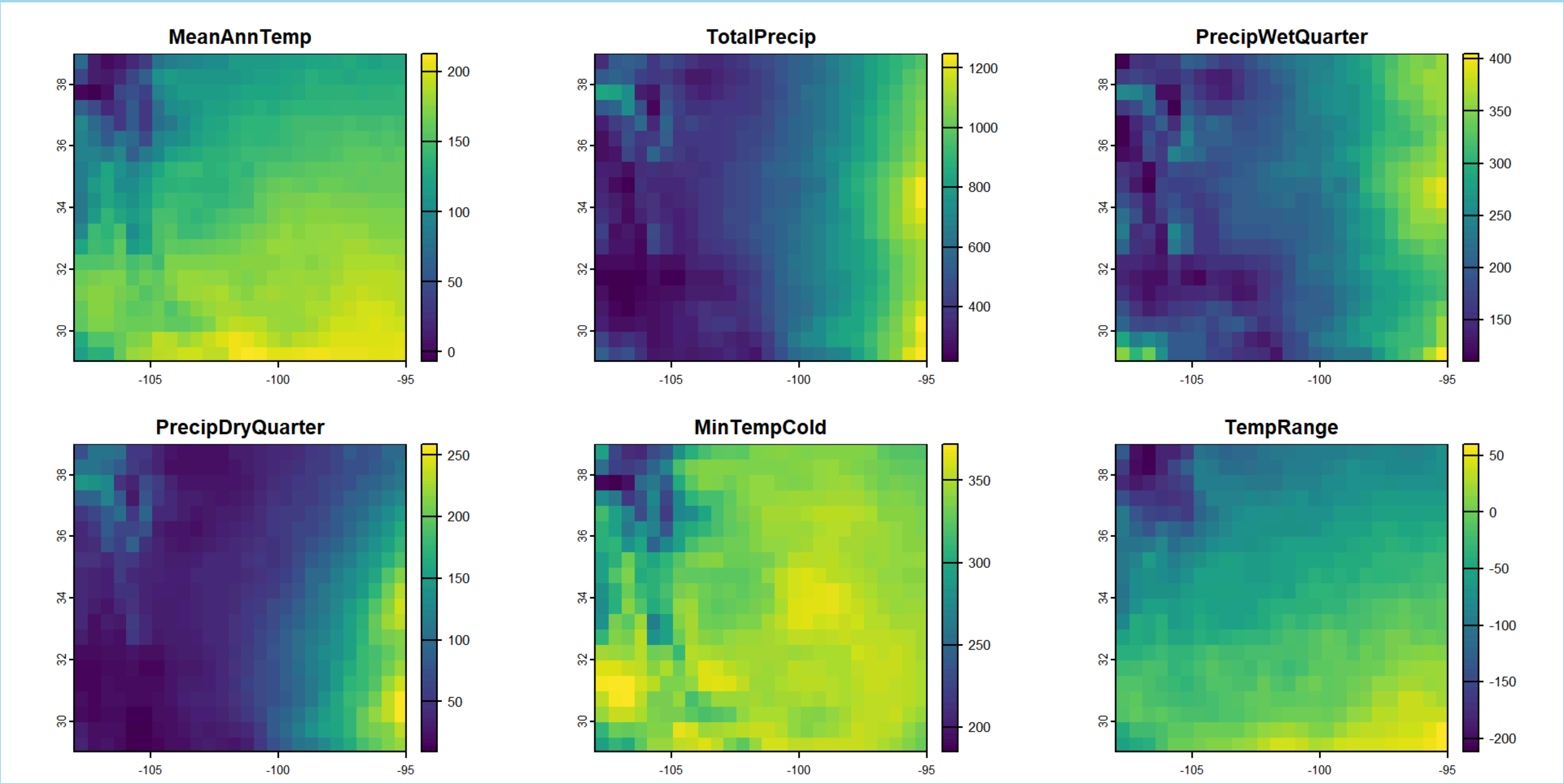
- Use decision rules to segment the predictor space
- Series of consecutive decision rules form a ‘tree’
- Terminal nodes (leaves) are the outcome; internal nodes (branches) the splits

Classification Trees

- Divide the predictor space () into non-overlapping regions
- Every observation in gets the same prediction
- *Recursive binary splitting*
- Pruning and over-fitting

An Example

Predictor inputs from the `dismo` package



An Example

Predictor inputs from the dismo package

```r
base.path <- "/opt/data/data/presabsexample/" #sets the path to the root di

pres.abs <- st_read(paste0(base.path, "presenceabsence.shp"), quiet = TRUE)
pred.files <- list.files(base.path,pattern='grd$', full.names=TRUE) #get th

pred.stack <- rast(pred.files) #read into a RasterStack
names(pred.stack) <- c("MeanAnnTemp", "TotalPrecip", "PrecipWetQuarter", "P
plot(pred.stack)
```

An Example

The sample data

An Example

Building our dataframe

```
pts.df <- terra::extract(pred.stack, vect(pres.abs), df=TRUE)
head(pts.df)
```

```
  ID MeanAnnTemp TotalPrecip PrecipWetQuarter PrecipDryQuarter MinTempCold
1  1         155         667              253               71         350
2  2         147         678              266               66         351
3  3         123         261              117               40         329
4  4         181         533              198               69         348
5  5         127         589              257               48         338
6  6          83         438              213               38         278
  TempRange
1       -45
2       -58
3       -64
4        -5
5       -81
6      -107
```

An Example

Building our dataframe

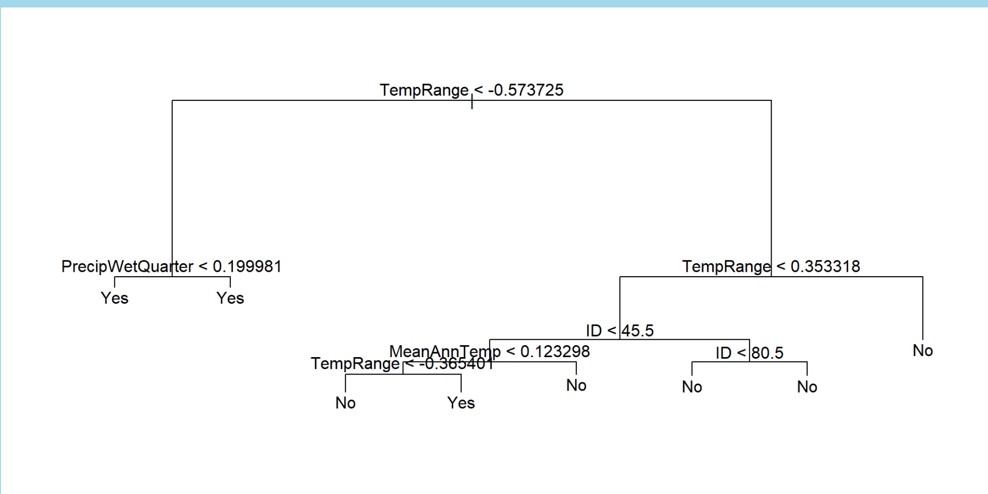
```
pts.df[,2:7] <- scale(pts.df[,2:7])
summary(pts.df)
```

```
       ID           MeanAnnTemp        TotalPrecip       PrecipWetQuarter
 Min.   :  1.00   Min.   :-3.3729   Min.   :-1.3377   Min.   :-1.6926
 1st Qu.: 25.75   1st Qu.:-0.4594   1st Qu.:-0.7980   1st Qu.:-0.6895
 Median : 50.50   Median : 0.2282   Median :-0.2373   Median :-0.2224
 Mean   : 50.50   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000
 3rd Qu.: 75.25   3rd Qu.: 0.7118   3rd Qu.: 0.7140   3rd Qu.: 0.6508
 Max.   :100.00   Max.   : 1.4285   Max.   : 2.4843   Max.   : 2.2713
 PrecipDryQuarter   MinTempCold         TempRange
 Min.   :-1.0828   Min.   :-3.9919   Min.   :-2.7924
 1st Qu.:-0.7013   1st Qu.:-0.0598   1st Qu.:-0.5216
 Median :-0.3770   Median : 0.3582   Median : 0.2075
 Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000
 3rd Qu.: 0.4290   3rd Qu.: 0.5495   3rd Qu.: 0.6450
 Max.   : 3.1713   Max.   : 1.1092   Max.   : 2.0407
```

An example

- Fitting the classification tree

```
library(tree)
pts.df <- cbind(pts.df, pres.abs$y)
colnames(pts.df)[8] <- "y"
pts.df$y <- as.factor(ifelse(pts.df$y == 1, "Yes", "No"))
tree.model <- tree(y ~ . , pts.df)
plot(tree.model)
text(tree.model, pretty=0)
```



An example

- Fitting the classification tree

```r
summary(tree.model)
```

```
Classification tree:
tree(formula = y ~ ., data = pts.df)
Variables actually used in tree construction:
[1] "TempRange"        "PrecipWetQuarter" "ID"               "MeanAnnTemp"
Number of terminal nodes:  8
Residual mean deviance:  0.3164 = 29.11 / 92
Misclassification error rate: 0.07 = 7 / 100
```

Benefits and drawbacks

Benefits

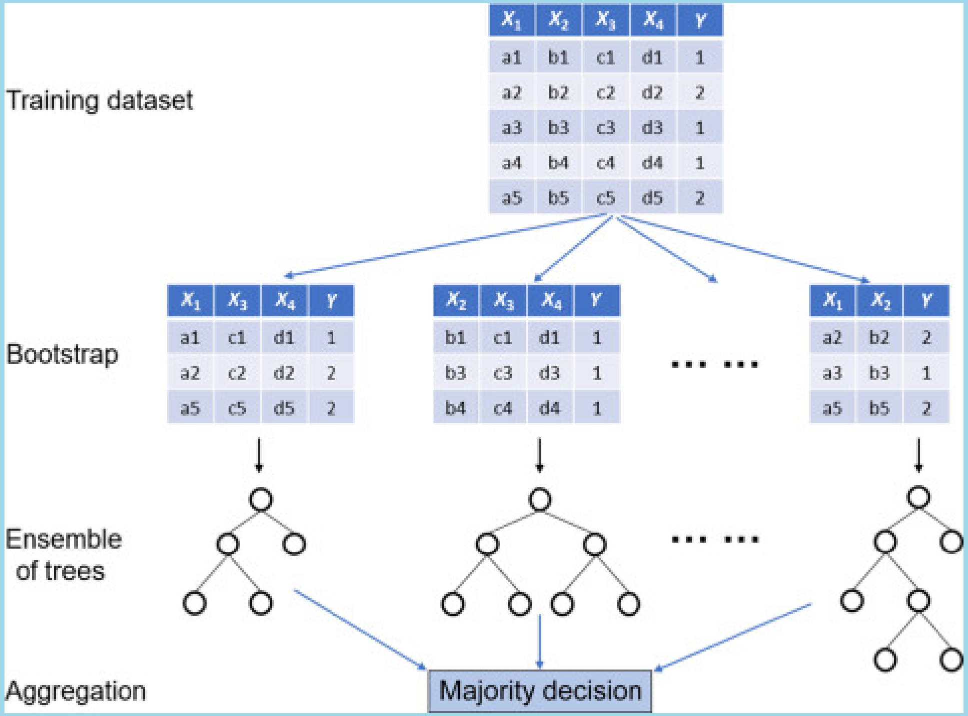
- Easy to explain
- Links to human decision-making
- Graphical displays
- Easy handling of qualitative predictors

Costs

- Lower predictive accuracy than other methods
- Not necessarily robust

Random Forests

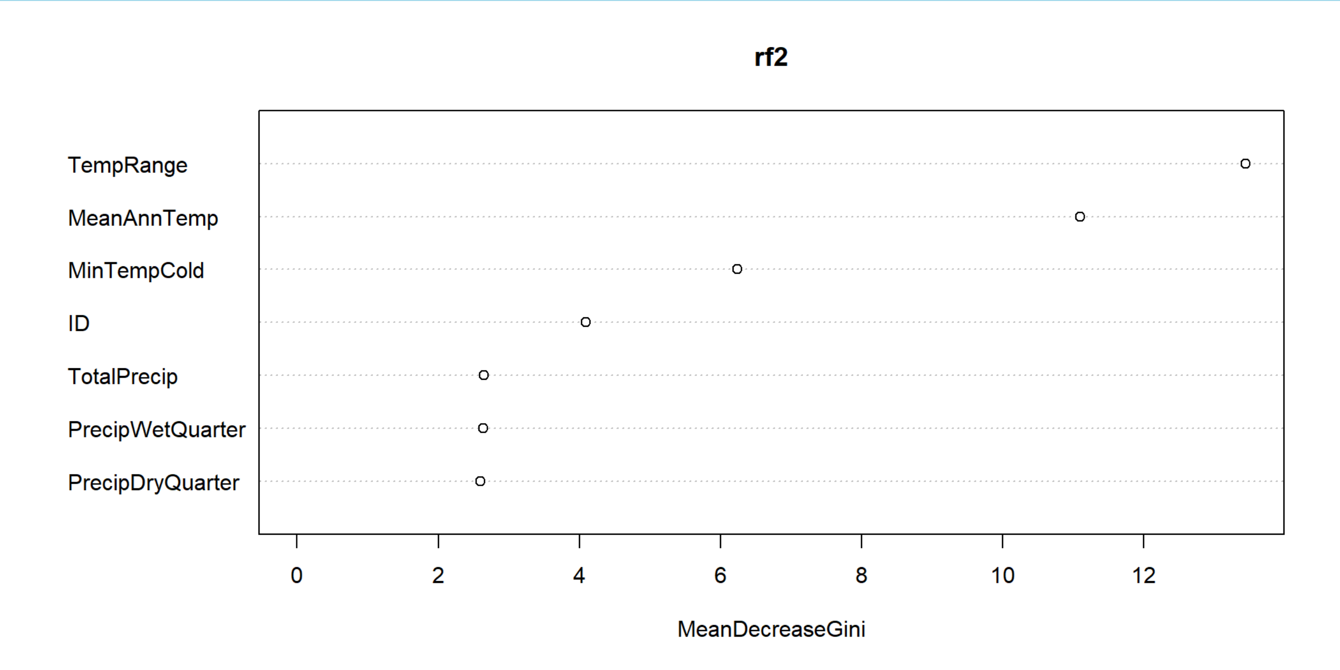
- Grow 100(000s) of trees using bootstrapping
- Random sample of predictors considered at each split
- Avoids correlation amongst multiple predictions
- Average of trees improves overall outcome (usually)
- Lots of extensions



An example

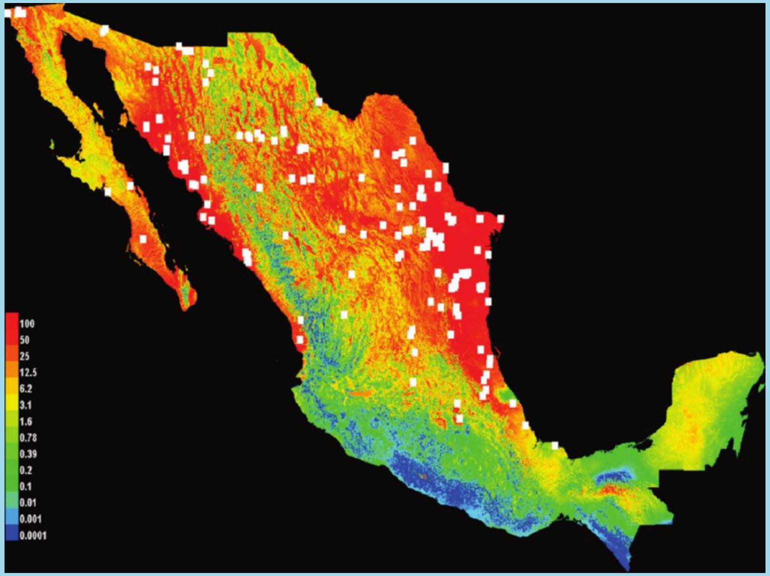
- Fitting the Random Forest

```
library(randomForest)
class.model <- y ~ .
rf2 <- randomForest(class.model, data=pts.df)
varImpPlot(rf2)
```



Modelling Presence-Background Data

The sampling situation



From Lentz et al. 2008

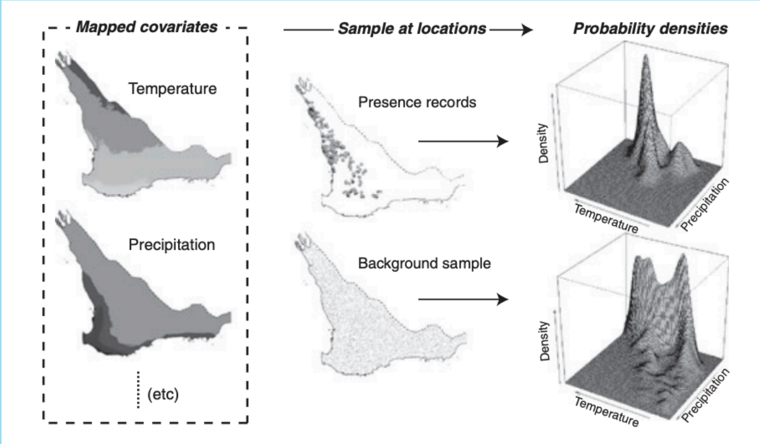
- Opportunistic collection of presences only
- Hypothesized predictors of occurrence are measured (or extracted) at each presence
- Background points (or pseudoabsences) generated for comparison

The Challenge with Background Points

- What constitutes background?
- Not measuring *probability*, but relative likelihood of occurrence
- Sampling bias affects estimation
- The intercept

Maximum Entropy models

- MaxEnt (after the original software)
- Need *plausible* background points across the remainder of the study area
- Iterative fitting to maximize the distance between predictions generated by a spatially uniform model
- Tuning parameters to account for differences in sampling effort, placement of background points, etc
- Development of the model beyond the scope of this course, but see Elith et al. 2010



From Elith et al. 2010

Challenges with MaxEnt

- Not measuring *probability*, but relative likelihood of occurrence
- Sampling bias affects estimation (but can be mitigated using tuning parameters)
- Theoretical issues with background points and the intercept
- Recent developments relate MaxEnt (with cloglog links) to Inhomogenous Point Process models

Extensions

- Polynomial, splines, piece-wise regression
- Neural nets, Support Vector Machines, many many more

Motivating Question

> How do Collaborative Forest Landscape Restoration projects compare to other National Forest lands with respect to social and wildfire risks?

Thinking about the data

- **Datasets** - Forest Service Boundaries, CFLRP Boundaries, Wildfire Risk Raster, CEJST shapefile
- **Dependent Variable** - CFLRP (T or F)
- **Independent Variables** - Wildfire hazard, income, education, housing burden

Building some Pseudocode

```
1. Load libraries
2. Load data
3. Check validity and alignment
4. Subset to relevant geographies
5. Select relevant attributes
6. Extract wildfire risk
7. CFLRP T or F
8. Compare risks
```

Load libraries

```r
library(sf)
library(terra)
library(tidyverse)
library(tmap)
```

Load the data

- Downloading USFS data using the function in the `code` folder

```
download_unzip_read <- function(link){
  tmp <- tempfile()
  download.file(link, tmp)
  tmp2 <- tempfile()
  unzip(zipfile=tmp, exdir=tmp2)
  shapefile.sf <- read_sf(tmp2)
}

### FS Boundaries
fs.url <- "https://data.fs.usda.gov/geodata/edw/edw_resources/shp/S_USA.Adm
fs.bdry <- download_unzip_read(link = fs.url)

### CFLRP Data
cflrp.url <- "https://data.fs.usda.gov/geodata/edw/edw_resources/shp/S_USA.
cflrp.bdry <- download_unzip_read(link = cflrp.url)
```