select_regression_predictors

2023-03-08

First, let's read in data.frames for annaul area burned and climate. Also import the 'car' library for regression diagnostic calculations

```
suppressPackageStartupMessages(library(car))
suppressPackageStartupMessages(library(data.table))

aab <- read.csv("../data/dataframes/annual_area_burned.csv")
cffdrs_vars <- read.csv("../data/dataframes/cffdrs-stats_era5_1979-2020.csv")
```

Now let's just check a quick summary for some of the variables in each of these datasets for the Alaskan ecoregion (50607).

```
summary(aab$annual_area_burned_km2[aab$ecos == 50607.0])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0   341.5  1319.0  3133.4  3469.5 18315.0
```

```
summary(cffdrs_vars$bui_95d[aab$ecos == 50607.0])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.491   9.526  16.242  18.160  24.557  52.906
```

```
summary(cffdrs_vars$isi_max[aab$ecos == 50607.0])
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -3.68080 -1.22300  0.02629  0.16767  1.49174  5.69301
```

```
# Merge datasets together using data.table
data_table <- data.table::merge.data.table(cffdrs_vars, aab,
                                           by = c("year", "ecos"))

# Rename "annual_area_burned_km2" -> "aab
colnames(data_table)[15] <- "aab"
```

Now, we want to create a list of model formulas that will be used for modeling aab in each ecoregion. This list of formulas will combine all possible bui and isi predictors so we can identify which specific combinations of explanatory variables should be used in a given ecoregion.

```
# Before we do this make sure to set any aab values < 4 km2 to a random uniform
# variable to help remove zeros.
id <- data_table$aab < 4
data_table$aab[id] <- runif(sum(id), 0, 4)

# Unique ecoregions
ecos = unique(data_table$ecos)

# Variables for CFFDDRS summaries:
# 1. max: Annual maximum anomaly relative 1980-2009
# 2. 95d: Number of days in a given year that exceed historical (1980-2009)
```

```r
#    95th percentile
# 3. fs: Max value from a 90-day moving window over each year
# 4. fwsl: The number of days in the fire season calculated from
#    Jolly et all 2015 Nature Comms
stats <- c("max", "95d", "fs", "fwsl")

mdl_forms <- c()

for (i in 1:length(stats)){
  for (j in 1:length(stats)){
    mdl_forms <- c(mdl_forms,paste0("aab ~ isi_", stats[i],
                                    " + bui_", stats[j]))
  }
}

# Print first four formulas to examine and make sure it's working
print(mdl_forms[1:4])
```

```
## [1] "aab ~ isi_max + bui_max"  "aab ~ isi_max + bui_95d"
## [3] "aab ~ isi_max + bui_fs"   "aab ~ isi_max + bui_fwsl"
```

The next step is to calculate a set of diagnostics for each set of explanatory variables, this includes AIC, BIC, Deviance, R-squared, and Variance Inflation Factor

```r
# Make data.frame to store results
diagnostics <- data.frame(ecos = numeric(),
                          mdl = character(),
                          AIC = numeric(),
                          BIC = numeric(),
                          Dev = numeric(),
                          R2 = numeric(),
                          vif = numeric())
```

In a nested for loop, go through and calculate each of the above diagnostics

```r
for (i in seq_along(ecos)){

  eco_id <- data_table$ecos == ecos[i]

  for (j in seq_along(mdl_forms)){

    glm_ij <- glm(mdl_forms[j],
                  family = Gamma(link = "log"),
                  data = data_table,
                  subset = eco_id)

    aic <- AIC(glm_ij)
    bic <- BIC(glm_ij)
    dev <- glm_ij$deviance
    dev.null <- glm_ij$null.deviance
    R2 <- 1 - dev / dev.null
    VIF <- car::vif(glm_ij)[1]

    df_ij <- data.frame(ecos = ecos[i],
                        mdl = mdl_forms[j],
                        AIC = aic,
```

```
                        BIC = bic,
                        Dev = dev,
                        R2 = R2,
                        vif = VIF)

    diagnostics <- rbind(diagnostics, df_ij, make.row.names = FALSE)

  }

}
```

Before we identify best models based on BIC and R2, remove models with high covariance among the predictors using VIF < 3 as a cutoff.

```
vif.id <- diagnostics$vif < 3
diagnostics <- diagnostics[vif.id, ]
```

In the following code, we fill in two empty vectors to hold the index of which models produced the lowest BIC and max R2 for each ecoregion.

```
min_bic_id <- numeric(length = length(ecos))

for (i in seq_along(ecos)){

  ecos.id <- which(diagnostics$ecos == ecos[i])
  bic.id <- which(diagnostics$BIC[ecos.id] == min(diagnostics$BIC[ecos.id]))

  min_bic_id[i] <- ecos.id[bic.id]

}

print(diagnostics[min_bic_id, ])
```

```
##      ecos                     mdl      AIC      BIC      Dev        R2
## 8   50602 aab ~ isi_95d + bui_fwsl 682.1707 689.1214 64.36877 0.2453642
## 17  50605  aab ~ isi_max + bui_max 566.8772 573.8278 84.83704 0.4668267
## 47  50606  aab ~ isi_fwsl + bui_fs 650.6431 657.5937 74.92835 0.3850060
## 51  50607   aab ~ isi_max + bui_fs 727.5290 734.4797 24.15015 0.6451976
## 65  50608  aab ~ isi_max + bui_max 713.2214 720.1721 53.27285 0.4215299
## 81  50609  aab ~ isi_max + bui_max 771.1774 778.1281 16.36043 0.6589442
## 107 50610    aab ~ isi_fs + bui_fs 628.7879 635.7386 83.97130 0.4891310
## 114 50612  aab ~ isi_max + bui_95d 731.9129 738.8636 36.78712 0.5587972
## 131 50613   aab ~ isi_max + bui_fs 595.7497 602.7004 76.34447 0.4111273
## 150 50614  aab ~ isi_95d + bui_95d 693.3521 700.3028 46.72102 0.4083286
## 165 50616  aab ~ isi_95d + bui_max 591.5839 598.5345 71.87658 0.4180263
## 179 51111   aab ~ isi_max + bui_fs 646.8580 653.8087 50.27782 0.4838409
##          vif
## 8   2.410950
## 17  1.797443
## 47  2.062715
## 51  1.768871
## 65  1.657674
## 81  2.096363
## 107 2.941773
## 114 1.555721
```

```
## 131 1.954968
## 150 1.945852
## 165 2.417577
## 179 1.596996
```

We now have an estimate of the best predictors to use for each ecoregion. We now want to export these results for use in the models

```r
write.csv(diagnostics[min_bic_id, ],
          "../data/dataframes/regression_predictors.csv",
          row.names = FALSE)
```