Package 'pollplot'

February 9, 2019

Title Functions for retrieving and plotting Norwegian polling results

Version 0.0.2

Description What the package does (one paragraph).

Depends R (>= 3.5.0), zoo

Imports stats, utils, graphics, grDevices, xts

License GPL (>= 2)

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

R topics documented:

election_results 1
get_polls 2

Index **4**

`election_results` *Norwegian general election results 1945–2017*

Description

Two matrices: `gen.votes` is the vote percentage each party received, `gen.mand` is the number of mandates each party received.

Usage

```
gen.votes
gen.mand
```

Format

An object of class `matrix` with 19 rows and 11 columns.

Examples

```
par(mfrow=c(3, 1), mar=c(1.8, 1.8, 1, 1), mgp=c(1.9, 0.6, 0),
  oma=c(0, 0, 2, 0), xaxs="i")

matplot(gen.votes[,1], gen.votes[,-1], type="o", pch=16, cex=0.9,
  lty=1, lwd=3, col=ppcol())
legend("top", "Percentage of votes", inset=-0.08,
  xpd=NA, bg="white", x.intersp=-0.5)
legend("top", colnames(gen.votes)[-1], inset=-0.28, horiz=TRUE, xpd=NA,
  x.intersp=0.5, bty="n", col=ppcol(), lwd=2.5, cex=0.85, seg.len=1)

gen.mand.p <- gen.mand
gen.mand.p[,-1] <- round(gen.mand[,-1]*100 / rowSums(gen.mand[,-1]), 2)

matplot(gen.mand.p[,1], gen.mand.p[,-1], type="o", pch=16, cex=0.9,
  lty=1, lwd=3, col=ppcol())
legend("top", "Percentage of mandates", inset=-0.08,
  xpd=NA, bg="white", x.intersp=-0.5)

gen.diff <- gen.mand[, -11]
gen.diff[,-1] <- gen.mand.p[,-c(1, 11)] - gen.votes[,-c(1, 11)]

matplot(gen.diff[,1], gen.diff[,-1], type="o", pch=16, cex=0.9,
  lty=1, lwd=3, col=ppcol())
legend("top", "Difference between %votes and %mandates", inset=-0.08,
  xpd=NA, bg="white", x.intersp=-0.5)
```

---

get_polls *Get polling results*

Description

Retrieve polling results from `pollofpolls.no`

Usage

```
get_polls(serieid = c(1:8, 15, 25, 160, 626), min.length = 8,
  verbose = TRUE, list = FALSE)
```

Arguments

`serieid`	id of the series to be retrieved
`min.length`	minimum length for included series
`verbose`	print progress messages
`list`	return output as a list

Details

Calls are made to `pollofpolls.no` to retrieve archived general election polling results.

Value

if `list=FALSE` an object of class `pollplot` is returned. This is an array of NA-padded polling data. Dates are along the first dimension, parties (always 10) are along the second dimension, and series are along the third dimension. In addition to dimnames, a few other attributes are included. `date` is a `"Date"` vector corresponding to the first dimension of the array. `info` is a `data.frame` of serieid, institute and client values corresponding to the retrieved series.

Examples

```
pp <- get_polls(5:8)
summary(pp)

# head, tail and window methods
# first year
head(pp, 365)
window(pp, start=start(pp), end=start(pp)+365)

# entire 2010
window(pp, start="2010-01-01", end="2010-12-31")

# last 30 days
tail(pp, 30)
window(pp, start=Sys.Date()-30)

# end is always Sys.Date() at the time of retrieval
end(pp)

# start is the date of the earliest sample. Depends on serieid
start(pp)
start(get_polls(10))

# basic plotting
pp.t <- window(pp, start=Sys.Date()-120)
pp.tf <- apply(pp.t, 1:2, mean, na.rm=TRUE)
nna <- which(!is.na(pp.tf[, 1]))
matplot(time(pp.t)[nna], pp.tf[nna, ], type="l", xaxt="n")
axis.Date(1, time(pp.t)[nna])
legend("top", colnames(pp.tf), ncol=5, bty="n", xpd=NA, inset=-0.15,
  cex=0.85, lty=1:5, col=1:6)

# more compact format. No methods for it...
pp.l <- get_polls(c(2, 7, 10), list=TRUE)

# but easily converted to a list of irregular-time zoo objects
library(zoo)
pp.z <- lapply(pp.l, function(x) read.zoo(x[,-(1:2)]))

library(xts)
pp.zc <- apply.daily(do.call(rbind.xts, pp.z), "mean")

plot(rollmean(pp.zc, 30))
```

Index

∗Topic **datasets**
election_results, 1

election_results, 1

gen.mand *(*election_results*)*, 1
gen.votes *(*election_results*)*, 1
get_polls, 2