MyDas

Stock prioritisation

Alex Tidd & Laurence Kell

15/03/18

Choice of stocks and justification

- The wiki has a summary stocks from the original call, however, the actual stocks still have to be chosen. This could be done based on a variety of justifications, e.g. commercial value of the stocks, ecological importance, or based on catch compostion of fleets. Stocks could also be selected based on productivity, i.e. stocks that are low in productivity relative to target species are more likely to be below B_{MSY}. Alternatively stocks could be selected based on susceptibility, i.e. are there found in the same area as the fishing metiers/fleets/gears.
- Preparation of data:

```r
library(DBI)
library(RPostgreSQL)
library(dplyr)
library(plyr)
library(reshape)
library(ggplot2)
library(RColorBrewer)
options(scipen = 999)


drv = dbDriver("PostgreSQL")

con = dbConnect(drv, host = "postgresql-seascope.csffkpr9jjjn.eu-west-2.rds.amazonaws.com",
    dbname = "mydasDev", port = 5432, user = "MydasApplication",
    password = "gmit2017!")


stecf = dbGetQuery(con, "SELECT * FROM data_stecflandings")
land2016 = subset(stecf, year %in% c(2008:2016))
# remove area 4bc etc
land2016$flag = ifelse(land2016$species %in% "LIN" & land2016$latitude <
    57.5 & land2016$area %in% "4", 1, 0)
land2016 = subset(land2016, flag == 0)
land2016 = subset(land2016, !(area %in% "6B RFMO"))
area = dbGetQuery(con, "SELECT * FROM div_area")

# ices division isnt broken down by division in North sea
land2016$division = ifelse(land2016$area %in% "4", "4A", land2016$division)
# convert in order to join and sum with landings
area$division = ifelse(area$division %in% c("12A", "12B", "12C"),
    "12", ifelse(area$division %in% c("14A", "14B"), "14", area$division))
# areas not in euro zone so remove
area = subset(area, !(area_27 %in% c("8.d.1", "7.k.1", "7.c.1",
    "6.b.1", "7.j.1")))
```

```r
totarea = ddply(area, .(division), summarise, totareakm = sum(area_km2))

allareas = inner_join(land2016, totarea)

allland = ddply(allareas, .(year, country, gear, stock, speciesgp),
    summarise, totland = sum(landings))
allland = subset(allland, totland > 0)
alllandarea = ddply(allareas, .(year, country, gear, stock, speciesgp,
    ices_rectangle), summarise, uniarea = unique(area_km2))
allareatot = ddply(alllandarea, .(year, country, gear, stock,
    speciesgp), summarise, totfleetarea = sum(uniarea))
combi = inner_join(allland, allareatot)

divarea = ddply(allareas, .(stock, division), summarise, area = unique(totareakm))
allstockarea = ddply(divarea, .(stock), summarise, stockarea = sum(area))

overlap = inner_join(combi, allstockarea)
overlap$olap_percent = (overlap$totfleetarea/overlap$stockarea) *
    100


catch = dbGetQuery(con, "SELECT * FROM data_stecf_aer_cpuedays")
catch$price = catch$totval/catch$totctch

price_sum = ddply(catch, .(year, speciesgp), summarise, price = mean(price))

overlap = inner_join(price_sum, overlap)
```

- Susceptibility

```r
# horizontal overlap categories 3 highest 1 lowest
overlap$score_olap = ifelse(overlap$olap_percent > 30, 3, ifelse(overlap$olap_percent >
    10 & overlap$olap_percent < 30, 2, ifelse(overlap$olap_percent <
    10, 1, 0)))
# price scoring ategories 3 highest 1 lowest
overlap$score_price = ifelse(overlap$price > 1.25, 3, ifelse(overlap$price >
    0.8 & overlap$price < 1.25, 2, ifelse(overlap$price < 0.8,
    1, 0)))

# catchability groupings 3 high, 2 medium 1 low
overlap$score_catch = ifelse(overlap$gear %in% c("BEAM") & overlap$speciesgp %in%
    c("BLL", "TUR", "GUG", "SKA"), 3, ifelse(overlap$gear %in%
    c("OTTER") & overlap$speciesgp %in% c("BLL", "TUR", "SKA",
    "JOD", "LIN", "POK", "POL"), 3, ifelse(overlap$gear %in%
    c("OTTER") & overlap$speciesgp %in% c("GUG"), 2, ifelse(overlap$gear %in%
    c("GILL") & overlap$speciesgp %in% c("POK", "POL"), 3, ifelse(overlap$gear %in%
    c("GILL") & overlap$speciesgp %in% c("LIN", "TUR"), 2, ifelse(overlap$gear %in%
    c("LONGLINE") & overlap$speciesgp %in% c("LIN"), 3, ifelse(overlap$gear %in%
    c("LONGLINE") & overlap$speciesgp %in% c("POL"), 2, ifelse(overlap$gear %in%
    c("PEL_TRAWL") & overlap$speciesgp %in% c("SPR"), 3, ifelse(overlap$gear %in%
    c("GILL") & overlap$speciesgp %in% c("POK", "POL"), 3, 1)))))))))


# Determination of susceptibility scores, adopted from Hobday
# et al. (2011) Evidence of post-capture release and survival
# =1, discarded but survivorship unknown =2, majority dead or
# retained =3
overlap$score_postc = ifelse(overlap$speciesgp %in% c("SKA",
    "POK"), 1, ifelse(overlap$speciesgp %in% c("BLL", "TUR",
    "POL", "LIN", "JOD"), 2, 3))

# calculate susceptibility

overlap$S = (((overlap$score_postc * overlap$score_olap * overlap$score_price *
    overlap$score_catch) - 1)/40) + 1
```

- Productivity

```
# 3 low productivity 1 high productivity < 5 years 5-15 years
# > 15 years
overlap$tm_score = ifelse(overlap$speciesgp %in% c("GUG", "POL",
    "POK", "TUR", "BLL", "SPR", "JOD"), 1, ifelse(overlap$speciesgp %in%
    c("SKA", "LIN"), 2, 3))
# > 20,000 eggs per year 100 - 20,000 eggs per year < 100
# eggs per year
overlap$fec_score = ifelse(overlap$speciesgp %in% c("SKA"), 3,
    ifelse(overlap$speciesgp %in% c("SPR"), 2, 1))
# Broadcast spawner Demersal egg layer Live bearer
overlap$repro_score = ifelse(overlap$speciesgp %in% c("SKA"),
    2, 1)
# < 2.75 2.75 - 3.25 > 3.25
overlap$troph_score = ifelse(overlap$speciesgp %in% c("SPR"),
    2, 3)
# < 40 cm 40-200 cm > 200 cm
overlap$lmat_score = ifelse(overlap$speciesgp %in% c("SPR", "BLL",
    "JOD", "GUG"), 1, 2)
# < 100 cm 100-300 cm > 300 cm
overlap$linf_score = ifelse(overlap$speciesgp %in% c("SPR", "BLL",
    "JOD", "GUG", "TUR", "POL"), 1, 2)
# calculate productivity
overlap$P = ((overlap$tm_score + overlap$fec_score + overlap$repro_score +
    overlap$troph_score + overlap$lmat_score + overlap$linf_score)/6)
```

[link]http://www.montereybayaquarium.org/-/m/C3EE8C68DA2A47B18A64BE6DBA72F76F.pdf

- Vulnerability

```
overlap$v = sqrt(overlap$P^2 + overlap$S^2)
```

- Ranking index giving equal Weight to the mean Value of landings and vulnerability

```
overlap$speciesgp = tolower(overlap$speciesgp)
overlap$value = overlap$price * overlap$totland * 1000
```

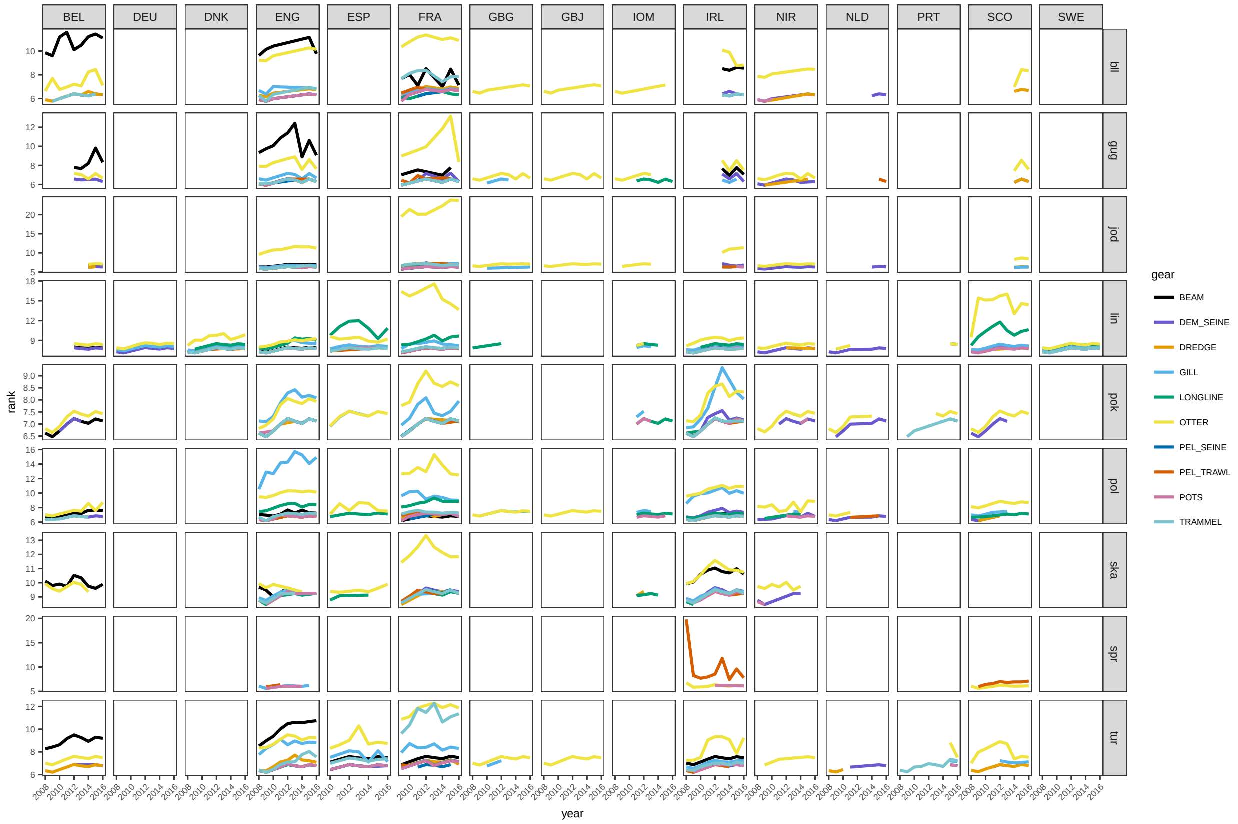
- Additive combination (ranking variable = scaled landings Value + scaled vulnerability) results in linear equal ranking lines over Value and vulnerability.

```
meanvals = ddply(overlap, .(year, country, gear, stock), summarise,
    mnval = mean(value))
totvals = ddply(meanvals, .(year), summarise, sdval = sd(mnval))
getv = inner_join(totvals, meanvals)
getv = na.omit(getv)
allv = inner_join(overlap, getv)
getvW = ddply(allv, .(year), summarise, sdv = sd(v))
allvs = inner_join(getvW, allv)

allvs$rank = (allvs$mnval/allvs$sdval) + (allvs$v/allvs$sdv)
rnkspc = allvs[c("year", "country", "gear", "speciesgp", "rank",
    "value", "sdv", "sdval", "v")]
```

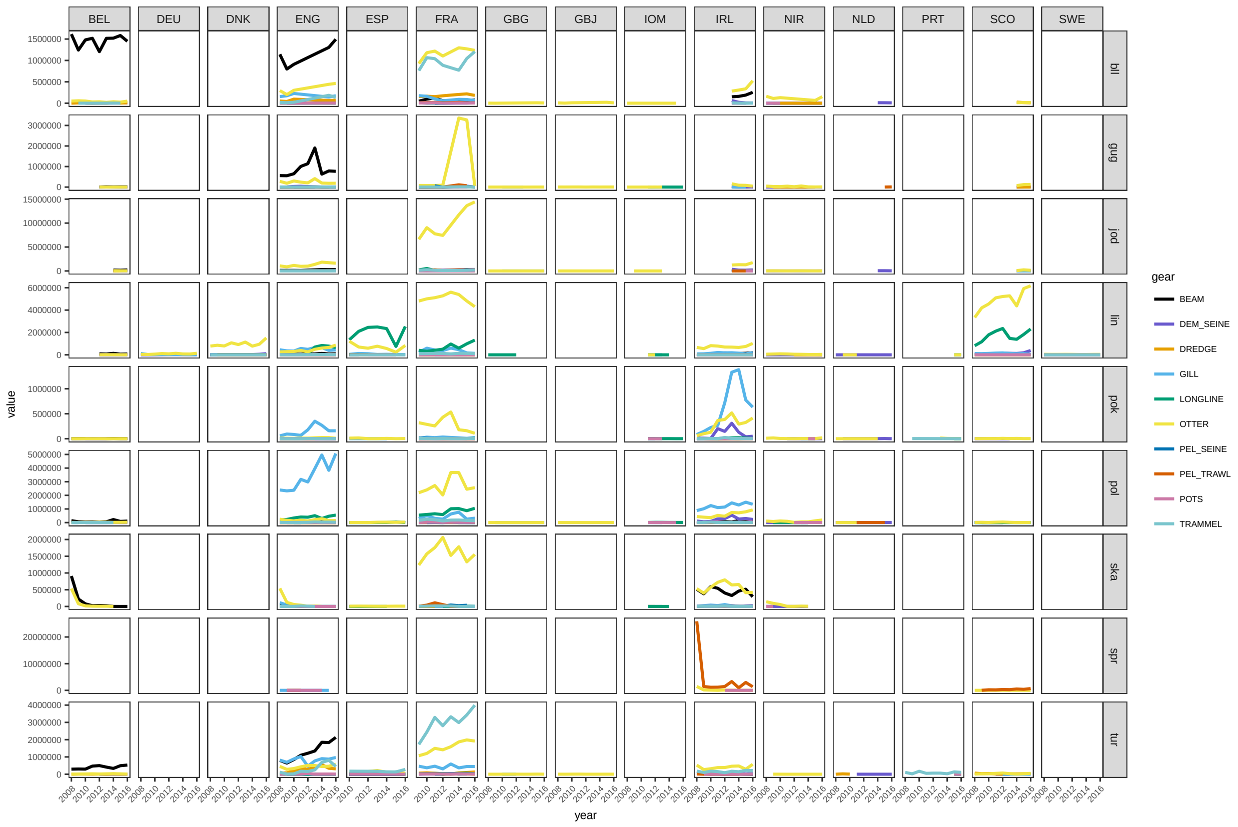
- Plot of ranking of variables by country, stock and gear

```
ggplot(subset(rnkspc, !(gear %in% "NONE")), aes(year, rank, group = gear,
    colour = gear)) + geom_line(size = 1) + facet_grid(speciesgp ~
    country, scales = "free") + theme_bw() + theme(text = element_text(size = 8),
    strip.text.x = element_text(size = 8), strip.text.y = element_text(size = 8),
    panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
    axis.text.x = element_text(angle = 45, hjust = 1)) + scale_color_manual(values = c("#000000",
    "slateblue3", "#E69F00", "#56B4E9", "#009E73", "#F0E442",
    "#0072B2", "#D55E00", "#CC79A7", "cadetblue3"))
```



- Plot of value (€) of variables by country, stock and gear

```
ggplot(subset(rnkspc, !(gear %in% "NONE")), aes(year, value,
    group = gear, colour = gear)) + geom_line(size = 1) + facet_grid(speciesgp ~
    country, scales = "free") + theme_bw() + theme(text = element_text(size = 8),
    strip.text.x = element_text(size = 8), strip.text.y = element_text(size = 8),
    panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
    axis.text.x = element_text(angle = 45, hjust = 1)) + scale_color_manual(values = c("#000000",
    "slateblue3", "#E69F00", "#56B4E9", "#009E73", "#F0E442",
    "#0072B2", "#D55E00", "#CC79A7", "cadetblue3"))
```

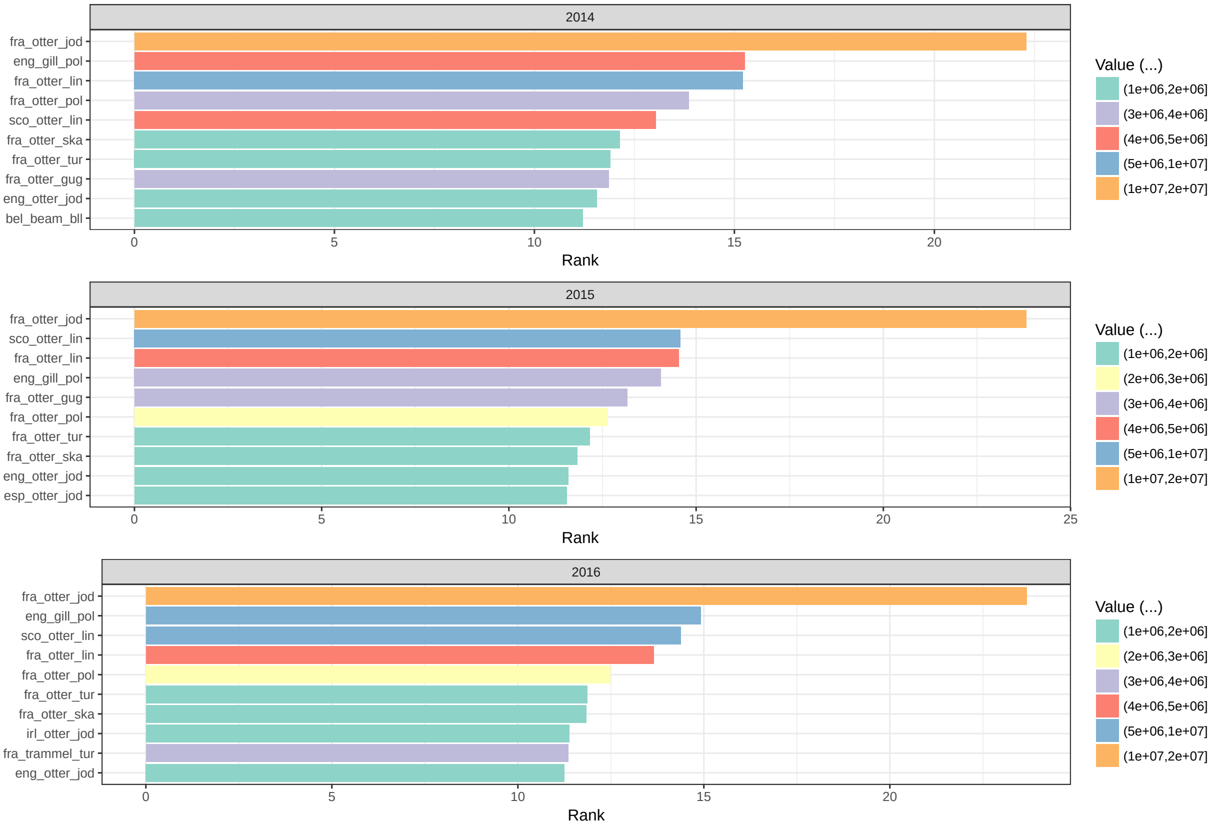


- ranking of the stocks by country and gear for last 3 years

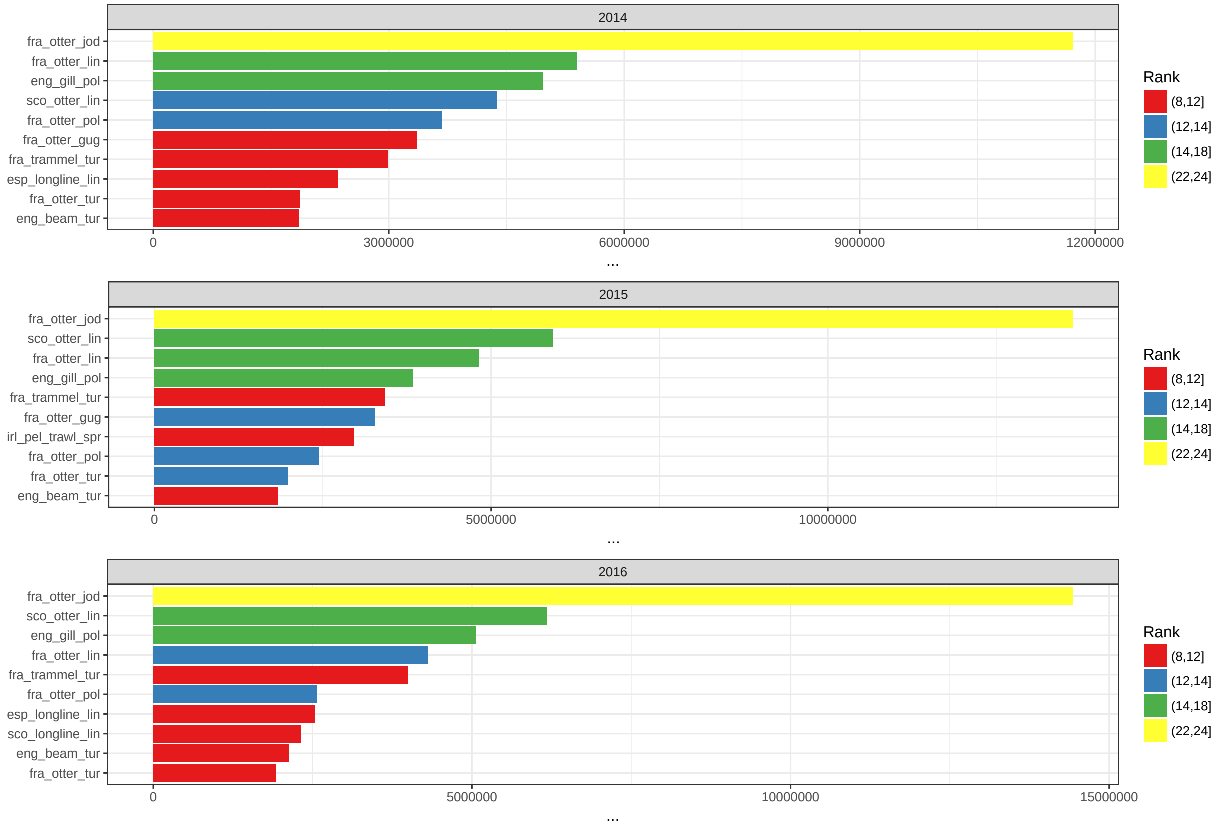
```
################ ranking
prioryr = subset(rnkspc, year %in% c(2014:2016))
prioryr$combo = paste(prioryr$country, prioryr$gear, prioryr$speciesgp,
    sep = "_")
tblrnk10 = prioryr %>% group_by(year) %>% arrange(desc(year),
    desc(rank), desc(value)) %>% top_n(10, rank)
tblrnk10 = tblrnk10 %>% group_by(year) %>% mutate(combo = factor(combo,
    levels = unique(combo)))
tblrnk10 = transform(tblrnk10, fill = cut(value, c(1000000, 2000000,
    3000000, 4000000, 5000000, 10000000, 20000000)))
mycolours = brewer.pal(6, "Set3")
names(mycolours) = levels(tblrnk10$fill)
# top 10 sorted on rank (combo of economics and
# vulnerability) and actual value
a = ggplot(subset(tblrnk10, year == 2014), aes(x = reorder(tolower(combo),
    rank), y = rank, fill = fill)) + geom_bar(stat = "identity") +
    coord_flip() + facet_wrap(~year, scale = "free_y") + guides(fill = guide_legend(title = "Value (€)"
    scale_fill_manual(name = "fill", values = mycolours) + theme_bw() +
    ylab("Rank") + xlab(" ")

b = ggplot(subset(tblrnk10, year == 2015), aes(x = reorder(tolower(combo),
    rank), y = rank, fill = fill)) + geom_bar(stat = "identity") +
    coord_flip() + facet_wrap(~year, scale = "free_y") + guides(fill = guide_legend(title = "Value (€)"
    scale_fill_manual(name = "fill", values = mycolours) + theme_bw() +
    ylab("Rank") + xlab(" ")

c = ggplot(subset(tblrnk10, year == 2016), aes(x = reorder(tolower(combo),
    rank), y = rank, fill = fill)) + geom_bar(stat = "identity") +
    coord_flip() + facet_wrap(~year, scale = "free_y") + guides(fill = guide_legend(title = "Value (€)"
    scale_fill_manual(name = "fill", values = mycolours) + theme_bw() +
    ylab("Rank") + xlab(" ")
gridExtra::grid.arrange(a, b, c, nrow = 3)
```



```
tblval10 = prioryr %>% group_by(year) %>% arrange(desc(year),
    desc(value), desc(rank)) %>% top_n(10, value)
tblval10 = transform(tblval10, fill = cut(rank, c(8, 12, 14,
    18, 20, 22, 24)))
tblval10 = tblval10 %>% group_by(year) %>% mutate(combo = factor(combo,
    levels = unique(combo)))
mycolours2 = brewer.pal(6, "Set1")
names(mycolours2) = levels(tblval10$fill)
# top 10 based purely on price and rank
d = ggplot(subset(tblval10, year == 2014), aes(x = reorder(tolower(combo),
    value), y = value, fill = fill)) + geom_bar(stat = "identity") +
    coord_flip() + facet_wrap(~year) + guides(fill = guide_legend(title = "Rank")) +
    scale_fill_manual(name = "fill", values = mycolours2) + theme_bw() +
    ylab("€") + xlab("")
e = ggplot(subset(tblval10, year == 2015), aes(x = reorder(tolower(combo),
    value), y = value, fill = fill)) + geom_bar(stat = "identity") +
    coord_flip() + facet_wrap(~year) + guides(fill = guide_legend(title = "Rank")) +
    scale_fill_manual(name = "fill", values = mycolours2) + theme_bw() +
    ylab("€") + xlab("")
f = ggplot(subset(tblval10, year == 2016), aes(x = reorder(tolower(combo),
    value), y = value, fill = fill)) + geom_bar(stat = "identity") +
    coord_flip() + facet_wrap(~year) + guides(fill = guide_legend(title = "Rank")) +
    scale_fill_manual(name = "fill", values = mycolours2) + theme_bw() +
    ylab("€") + xlab("")
gridExtra::grid.arrange(d, e, f, nrow = 3)
```


fra_otter_jod
fra_otter_lin
eng_gill_pol
sco_otter_lin
fra_otter_pol
fra_otter_gug
fra_trammel_tur
esp_longline_lin
fra_otter_tur
eng_beam_tur
Rank
(8,12]
(12,14]
(14,18]
(22,24]
fra_otter_jod
sco_otter_lin
fra_otter_lin
eng_gill_pol
fra_trammel_tur
fra_otter_gug
irl_pel_trawl_spr
fra_otter_pol
fra_otter_tur
eng_beam_tur
Rank
(8,12]
(12,14]
(14,18]
(22,24]
fra_otter_jod
sco_otter_lin
eng_gill_pol
fra_otter_lin
fra_trammel_tur
fra_otter_pol
esp_longline_lin
sco_longline_lin
eng_beam_tur
fra_otter_tur
Rank
(8,12]
(12,14]
(14,18]
(22,24]