Carbon

Vincent Bisquay Gracia

12/05/2021

I. set up

Used pakages :

```
setwd(
  "D:/Users/181248/Documents/R")
library(magrittr)
library(cowplot)
library(ggplot2)
library(raster)
library(animation)
library(rasterVis)
library(knitr)
library(tibble)
library(dplyr)
```

```
BAU_input =
  "D:/Users/181248/Documents/Klamath_(CA_only)_2021 - Harvest BAU/"
Adapty_input =
  "D:/Users/181248/Documents/Klamath_(CA_only)_2021 - Harvest Adaptability/"
ProAct_input =
  "D:/Users/181248/Documents/Klamath_(CA_only)_2021 - Harvest Pro-active/"

BAU_inputBM =
  "D:/Users/181248/Documents/Klamath_(CA_only)_2021 - Harvest BAU/..1/biomass/"
Adapty_inputBM =
  "D:/Users/181248/Documents/Klamath_(CA_only)_2021 - Harvest Adaptability/..1/biomass/"
ProAct_inputBM =
  "D:/Users/181248/Documents/Klamath_(CA_only)_2021 - Harvest Pro-active/..1/biomass/"

ListScenario = c("BAU","Adapty","ProAct")
```

```
pal.2 = colorRampPalette(c("purple","yellow"))

pal.dif = colorRampPalette(c("red","tomato", "darkgoldenrod1","cyan3","cyan","blue"))
```

II. Carbon balance

III. Resilience and disturbancies regimes

The biomass killed by fires coming from lightnings (natural fire mortality), by fires coming from human activity (called here accidental fires, even if they can be intentionnal) and by insects (insect mortality) is compared on the different scenarios. To avoid an overestimation of mortality and to make sure that all the data frames havethe same length, the years without insect outbreak are added with a mortality of 0.

III.a) Mortality due to disturbancies

```
Mortality = data.frame()

for(scenario in ListScenario){

  if(scenario == "BAU") {Input = BAU_input
  } else if(scenario == "Adapty"){Input = Adapty_input
  } else {Input = ProAct_input}

  BDA_Tot = paste(Input,"..1/bda_log.csv",sep="") %>% read.csv()
  BDA = cbind(BDA_Tot$Time, BDA_Tot$TotalBiomassMortality) %>% as.data.frame()
  colnames(BDA) = c("Time", "Mortality")

  BDA = aggregate(BDA$Mortality, by=list(Time=BDA$Time), FUN=sum)

  colnames(BDA) = c("Time", "Mortality")


  for(i in (1:100)){
    if(!(i %in% BDA$Time)){
    BDA = rbind(BDA, c(i,0))
    }
  }

  BDA$Disturbancy = "Insects"
  BDA$Scenario = scenario

  scrapple_Tot = paste(Input, "..1/scrapple-summary-log.csv", sep="") %>% read.csv()
  scrapple_Acc = cbind(scrapple_Tot$SimulationYear,
                scrapple_Tot$TotalBiomassMortalityAccidental) %>% as.data.frame()
  colnames(scrapple_Acc) = c("Time", "Mortality")
  scrapple_Acc$Disturbancy = "Fire accidental"
  scrapple_Acc$Scenario = scenario

  scrapple_Nat = cbind(scrapple_Tot$SimulationYear,
                scrapple_Tot$TotalBiomassMortalityLightning) %>% as.data.frame()
  colnames(scrapple_Nat) = c("Time", "Mortality")
  scrapple_Nat$Disturbancy = "Fire lightning"
  scrapple_Nat$Scenario = scenario

  Harvest_Tot =  paste(Input,"..1/harvest-event-test-log.csv",sep="") %>% read.csv()
  Harvest = Harvest_Tot %>% group_by(Time) %>%
    summarise(MgBiomassRemoved = sum(MgBiomassRemoved))
  colnames(Harvest) = c("Time", "Mortality")
```

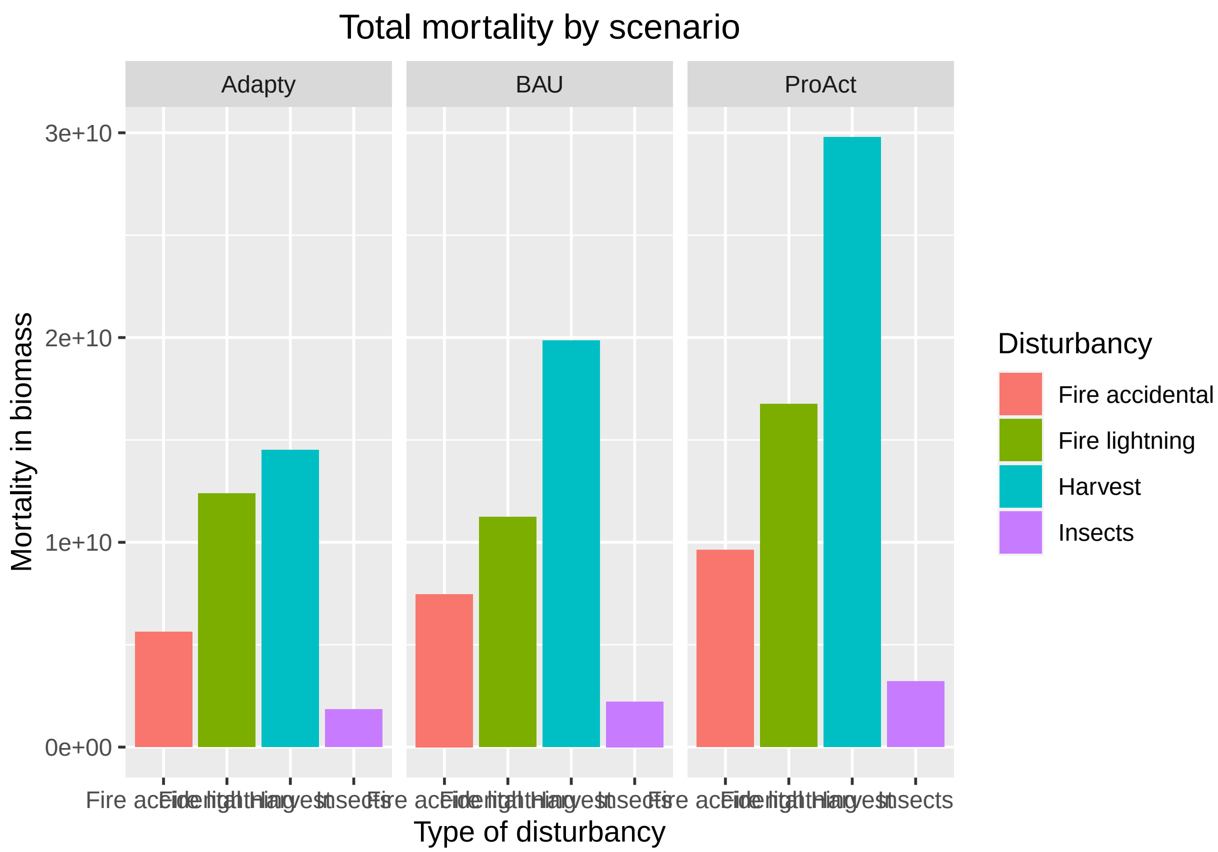
```r
  Harvest$Mortality = Harvest$Mortality %>% as.numeric()
  Harvest$Mortality = Harvest$Mortality*100 # convertion Mg/ha in g/m²
  Harvest$Disturbancy = "Harvest"
  Harvest$Scenario = scenario


  Mortality = rbind(Mortality, BDA, scrapple_Acc, scrapple_Nat, Harvest)
}
```

The biomass removed by harvest is also added as a mortality.

```r
ggplot(Mortality, aes(x=Time, y=Mortality, col=Disturbancy, group=Disturbancy)) +
  geom_line()+
#  geom_area(aes(fill=Disturbancy)) +
  xlab("Time (years)") +
  ylab("Mortality in biomass") +
  ggtitle("Evolution of mortality due to disturbancies on different scenarios") +
  theme(plot.title = element_text(hjust = 0.5))+
  facet_wrap(~Scenario, ncol=1)
```


```r
ggplot(Mortality, aes(x=Disturbancy, y=Mortality, fill = Disturbancy)) +
  geom_bar(stat="identity") +
  xlab("Type of disturbancy") +
  ylab("Mortality in biomass") +
  ggtitle("Total mortality by scenario") +
```

```
  theme(plot.title = element_text(hjust = 0.5))+
  facet_wrap(~Scenario, ncol=3)
```



Non parametric test and paired => Wilcoxon Test

```
MorTable = c("Scenario", "Disturbancy", "mean_mortality", "standard_error")
TestData = data.frame()

for (disturbance in c("Insects","Fire accidental", "Fire lightning", "Harvest")) {
  for (scenario in ListScenario) {
    Mort = Mortality$Mortality[Mortality$Disturbancy==disturbance &
                               Mortality$Scenario==scenario] %>%
                                as.numeric() %>% as.data.frame()
    MeanMort = mean(Mort$.)
    sdMort = sd(Mort$.)

    MorTable = rbind(MorTable, c(scenario, disturbance, MeanMort, sdMort))

    names(Mort) = paste0(disturbance, "_", scenario)

    if (nrow(TestData) == 0) {
      TestData = Mort
    } else {
      TestData = cbind(TestData, Mort)
    }
  }
```

```r
}

colnames(MorTable)=MorTable[1,] %>% unlist %>% as.character
MorTable=MorTable[-1,] %>% as.data.frame()


kable(as.data.frame(MorTable), caption = "Mortality depending on the scenario")
```

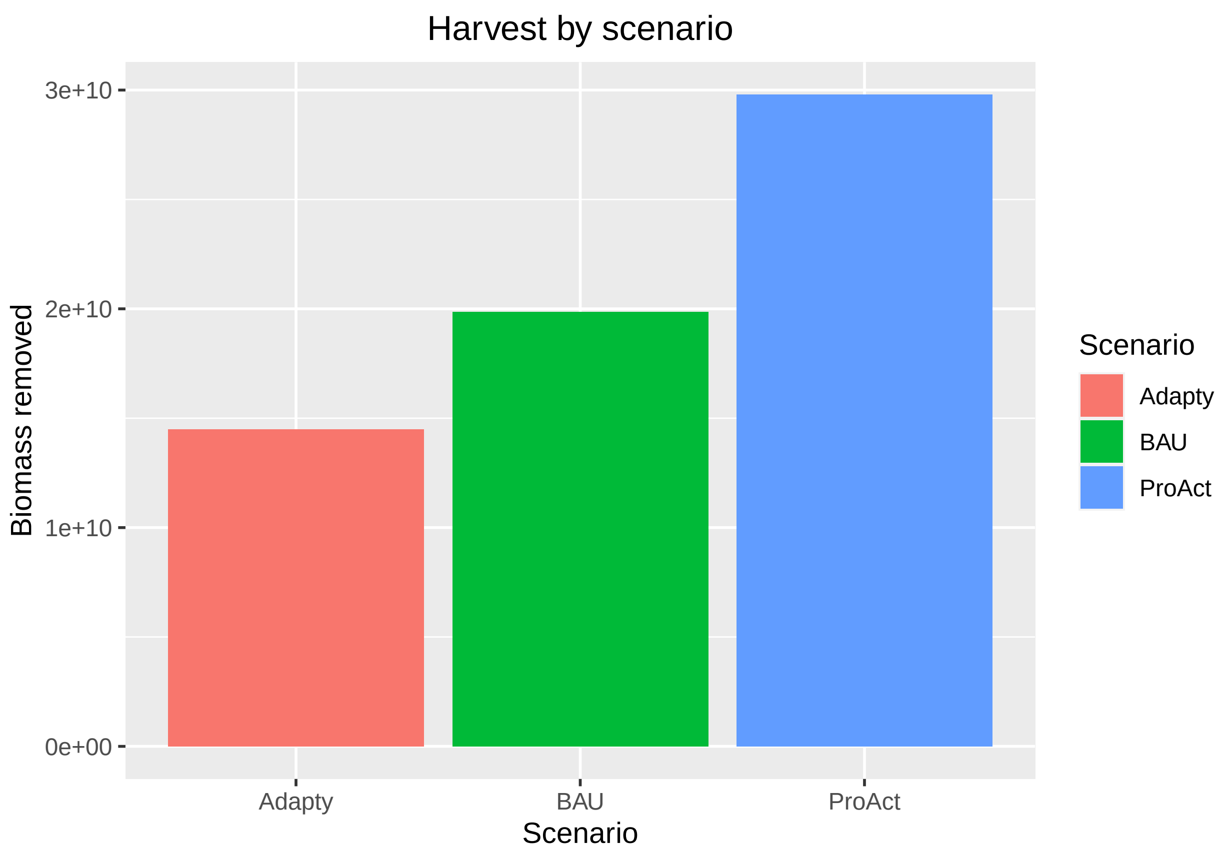
Table 1: Mortality depending on the scenario

	Scenario	Disturbancy	mean_mortality	standard_error
X	BAU	Insects	22168807.25	13582818.7063869
X.1	Adapty	Insects	18558547.89	13534102.4493461
X.2	ProAct	Insects	32060499.05	19781696.0456064
X.3	BAU	Fire accidental	74634314.05	133961575.396683
X.4	Adapty	Fire accidental	56312642.36	114925948.294583
X.5	ProAct	Fire accidental	96213709.29	168079385.959997
X.6	BAU	Fire lightning	112344476.04	157696468.161566
X.7	Adapty	Fire lightning	124046170.44	190034967.165735
X.8	ProAct	Fire lightning	167550640.98	258488580.382901
X.9	BAU	Harvest	993184160.1	203520538.089187
X.10	Adapty	Harvest	725060866.7	106260625.229878
X.11	ProAct	Harvest	1489218233.6	281065840.760762

```r
#MorTable$mean_mortality = MorTable$mean_mortality %>% as.numeric()
#MorTable$sd = MorTable$sd %>% as.numeric()

#MorTable$mean_mortality = MorTable$mean_mortality %>% formatC(format="e", digits = 2)
#MorTable$sd = MorTable$sd %>% formatC(format="e", digits = 2)

#MorTable$Table = paste0(MorTable$mean_mortality, "(",MorTable$sd,")")
```

III.a.1) Harvest

```r
ggplot(Mortality[Mortality$Disturbancy=="Harvest",], aes(x=Scenario, y=Mortality, fill=Scenario)) +
geom_bar(stat="identity") +
ylab("Biomass removed") +
ggtitle("Harvest by scenario") +
theme(plot.title = element_text(hjust = 0.5))
```




```r
shapiro.test(TestData$Harvest_BAU)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  TestData$Harvest_BAU
## W = 0.89463, p-value = 7.976e-07
```

```r
shapiro.test(TestData$Harvest_Adapty)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  TestData$Harvest_Adapty
## W = 0.79967, p-value = 2.419e-10
```

```r
shapiro.test(TestData$Harvest_ProAct)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  TestData$Harvest_ProAct
## W = 0.86989, p-value = 6.938e-08
```

```r
wilcox.test(TestData$Harvest_BAU, TestData$Harvest_Adapty)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  TestData$Harvest_BAU and TestData$Harvest_Adapty
## W = 9050, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(TestData$Harvest_BAU, TestData$Harvest_ProAct)
```

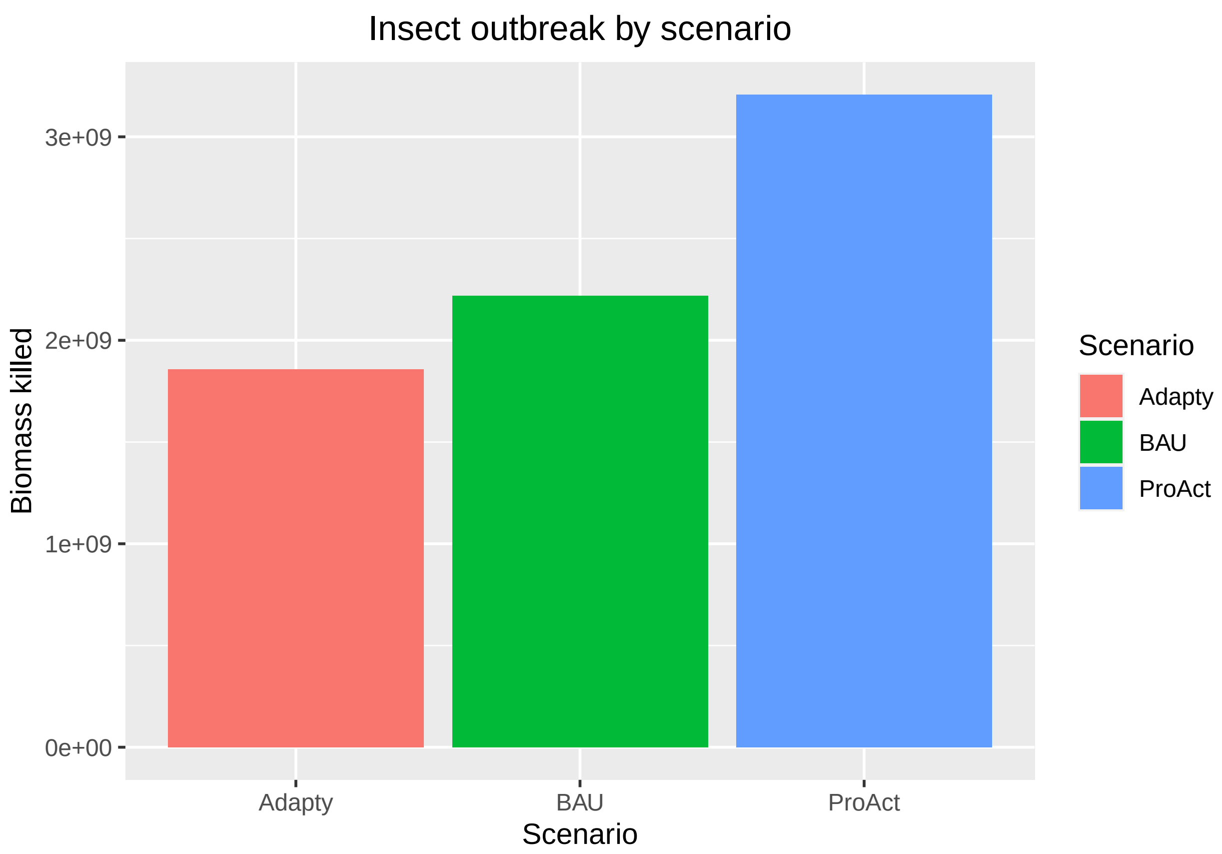
```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  TestData$Harvest_BAU and TestData$Harvest_ProAct
## W = 900, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(TestData$Harvest_Adapty, TestData$Harvest_ProAct)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  TestData$Harvest_Adapty and TestData$Harvest_ProAct
## W = 50, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

III.b.2) Insects

```r
ggplot(Mortality[Mortality$Disturbancy=="Insects",], aes(x=Scenario, y=Mortality, fill=Scenario)) +
geom_bar(stat="identity") +
ylab("Biomass killed") +
ggtitle("Insect outbreak by scenario") +
theme(plot.title = element_text(hjust = 0.5))
```



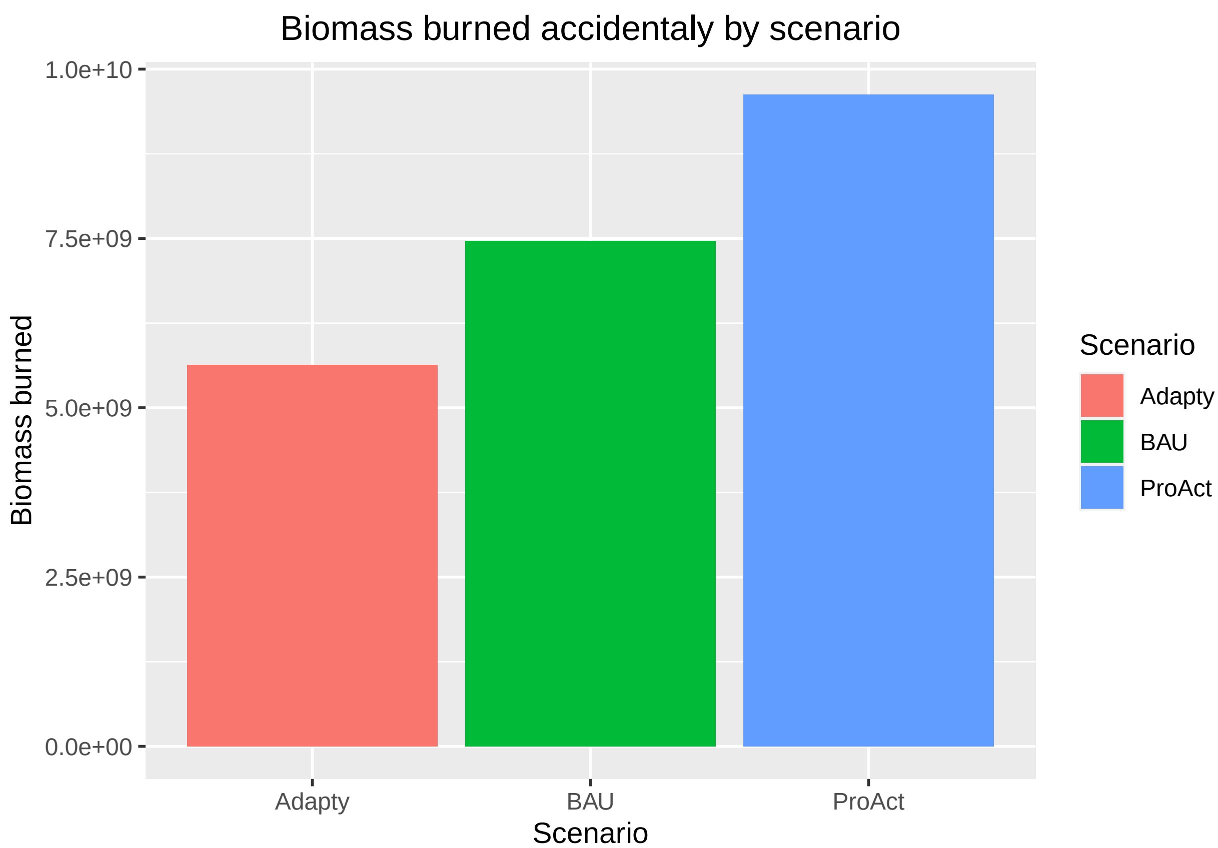

```
#kable(as.data.frame(cbind(MorTable$Scenario[MorTable$Disturbancy=="Insects"],
#                           MorTable$Table[MorTable$Disturbancy=="Insects"])))
```

III.c.3) Fire

i) Accidental

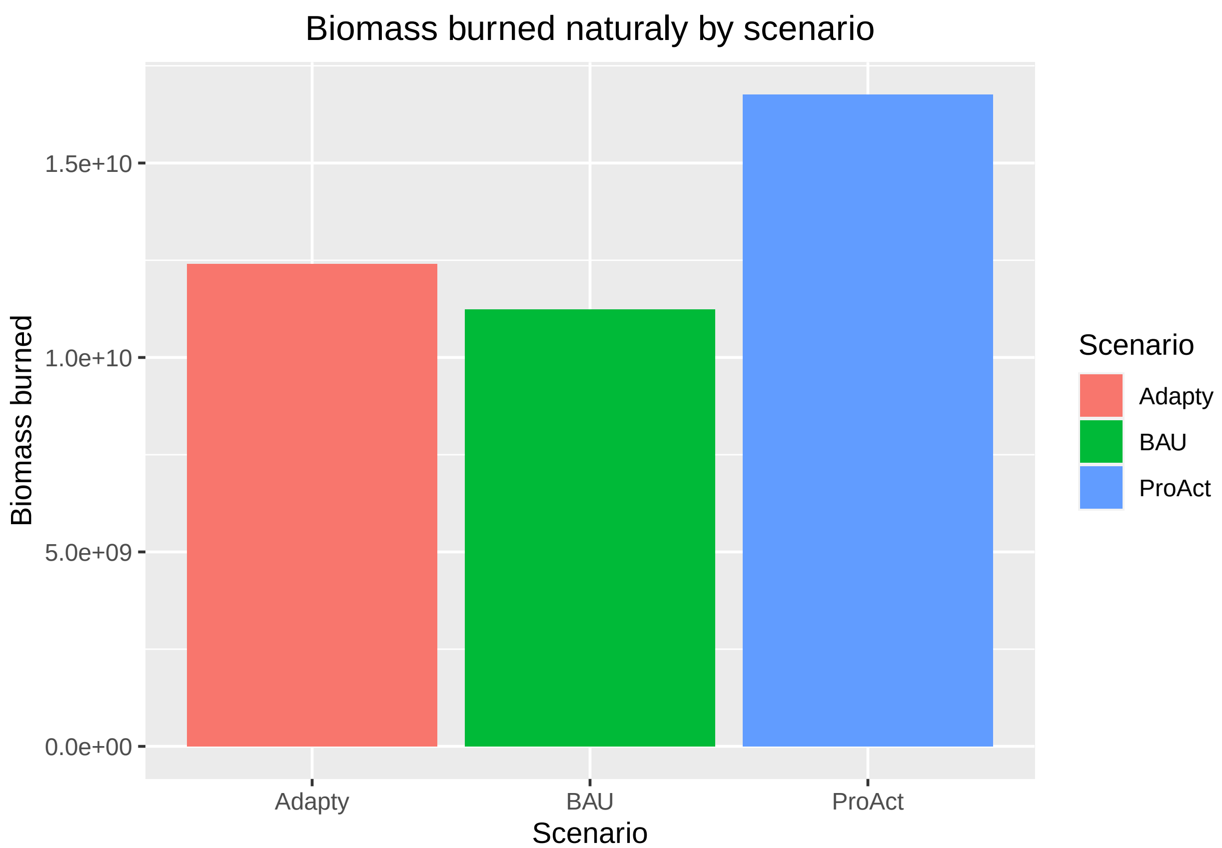
Don't consider the Rx here

```
ggplot(Mortality[Mortality$Disturbancy=="Fire accidental",], aes(x=Scenario, y=Mortality, fill=Scenario)
geom_bar(stat="identity") +
ylab("Biomass burned") +
ggtitle("Biomass burned accidentaly by scenario") +
theme(plot.title = element_text(hjust = 0.5))
```



ii) Natural

```
ggplot(Mortality[Mortality$Disturbancy=="Fire lightning",], aes(x=Scenario, y=Mortality, fill=Scenario)
geom_bar(stat="identity") +
ylab("Biomass burned") +
ggtitle("Biomass burned naturaly by scenario") +
theme(plot.title = element_text(hjust = 0.5))
```



Mean mortality, max, f pics

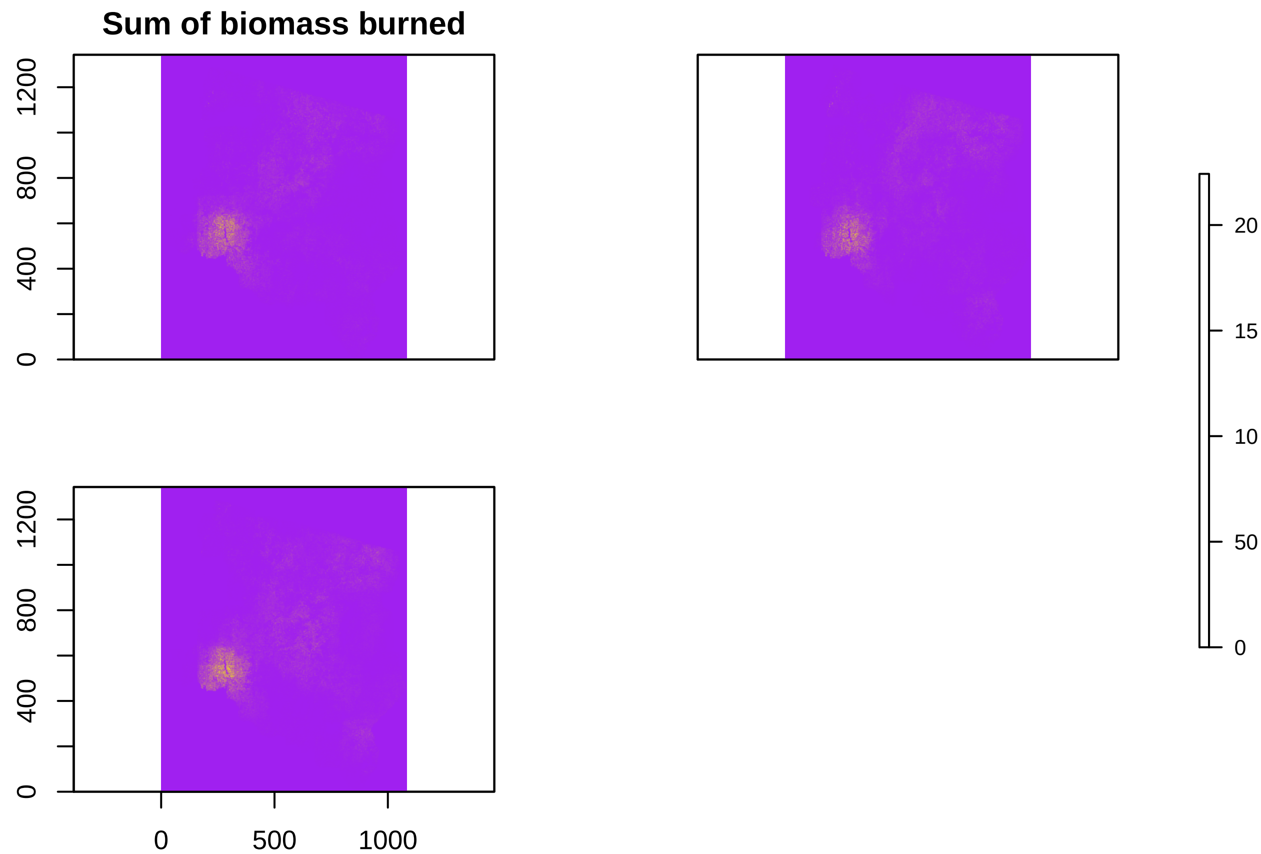
biomass recover after disturbancy (slope).

```
Zero = paste0(BAU_input, "..1/scrapple-fire/flaming-consumptions-1.img") %>% raster()

Zero[Zero != 0] = 0

RasterStack_Burn = stack()
RasterStack_AllF = stack()
RasterStack_AccF = stack()
RasterStack_NatF = stack()
RasterStack_RxF = stack()

for(scenario in ListScenario){

  if(scenario == "BAU") {
    Input = BAU_input
    TotBurn = Zero
    FreqAllT = Zero
    FreqAcc = Zero
    FreqNat = Zero
    FreqRx = Zero
  } else if(scenario == "Adapty"){
    Input = Adapty_input
    TotBurn = Zero
    TotBurn = Zero
```

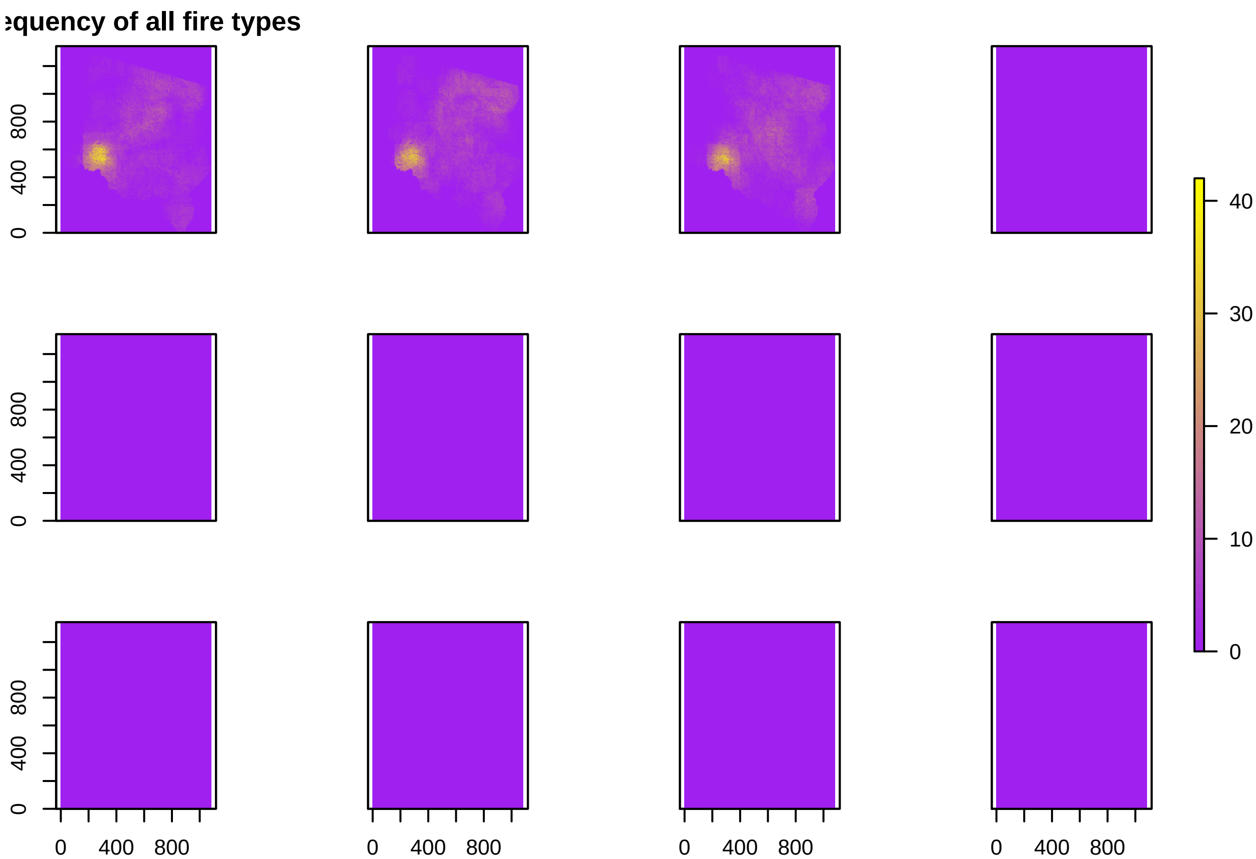
```r
  FreqAllT = Zero
  FreqAcc = Zero
  FreqNat = Zero
  FreqRx = Zero
} else {
  Input = ProAct_input
  TotBurn = Zero
  TotBurn = Zero
  FreqAllT = Zero
  FreqAcc = Zero
  FreqNat = Zero
  FreqRx = Zero}

for (i in 1:100) {
  Raster_fc = paste0(Input, "..1/scrapple-fire/flaming-consumptions-",i,".img") %>% raster()
  Raster_sm = paste0(Input, "..1/scrapple-fire/smolder-consumption-",i,".img") %>% raster()
  RasterBurn = Raster_fc + Raster_sm
  TotBurn = TotBurn + RasterBurn

  Raster_FT = paste0(Input, "..1/scrapple-fire/ignition-type-",i,".img") %>% raster()

  Raster_FT[Raster_FT == 1] = 0

  Raster_AllT = Raster_FT
  Raster_FT[Raster_FT != 0] = 1
  FreqAllT = FreqAllT + Raster_FT

  Raster_Acc = Raster_FT
  Raster_Acc[Raster_Acc != 2] = 0
  Raster_Acc[Raster_Acc == 2] = 1
  FreqAcc = FreqAcc + Raster_Acc

  Raster_Nat = Raster_FT
  Raster_Nat[Raster_Nat != 3] = 0
  Raster_Nat[Raster_Nat == 3] = 1
  FreqNat = FreqNat + Raster_Nat

  Raster_Rx = Raster_FT
  Raster_Rx[Raster_Rx != 4] = 0
  Raster_Rx[Raster_Rx == 4] = 1
  FreqRx = FreqRx + Raster_Rx

  if(i==100){
    #TotBurn[TotBurn == 0] = NA
    names(TotBurn) = scenario
    RasterStack_Burn = stack(RasterStack_Burn, TotBurn)

    #FreqAllT[FreqAllT == 0] = NA
    names(FreqAllT) = paste0(scenario,"all")
    RasterStack_AllF = stack(RasterStack_AllF, FreqAllT)

    #FreqAcc[FreqAcc == 0] = NA
    names(FreqAcc) = paste0(scenario,"Acc")
```

```
      RasterStack_AccF = stack(RasterStack_AccF, FreqAcc)

      #FreqNat[FreqNat == 0] = NA
      names(Raster_Nat) = paste0(scenario,"Nat")
      RasterStack_NatF = stack(RasterStack_NatF, FreqNat)

      #FreqRx[FreqRx == 0] = NA
      names(FreqRx) = paste0(scenario,"Rx")
      RasterStack_RxF = stack(RasterStack_RxF, FreqRx)

    }
      }
}
```

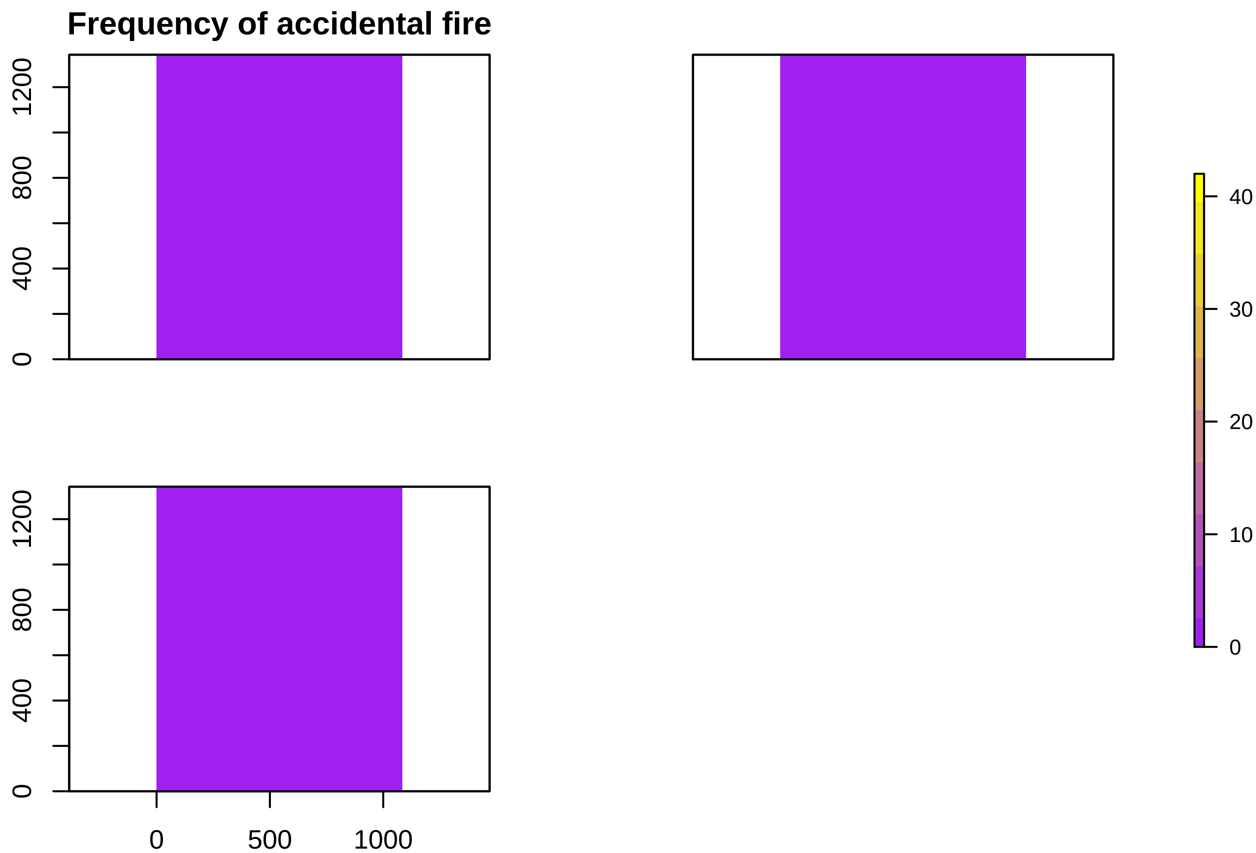
```
VarSatck = RasterStack_Burn
  Max = max(maxValue(VarSatck))+1
  Min = min(minValue(VarSatck))
  brks = seq(Min,Max,by=0.1)
  nbrks = length(brks)-1
  r.range = c(Min, Max)

rasTot = raster(ncol=2, nrow=1)
values(rasTot) = c(Min,Max)

par(mfrow=c(1,3))

plot(VarSatck, breaks=brks,col=pal.2(nbrks), legend = F, zlim=c(Min,Max),
main = "Sum of biomass burned")
plot(rasTot, legend.only=TRUE, col=pal.2(nbrks),
    legend.width=1, legend.shrink=0.75,
    legend.args=list(text="??",
                     side=4, font=2, line=2.5, cex=0.8))
```



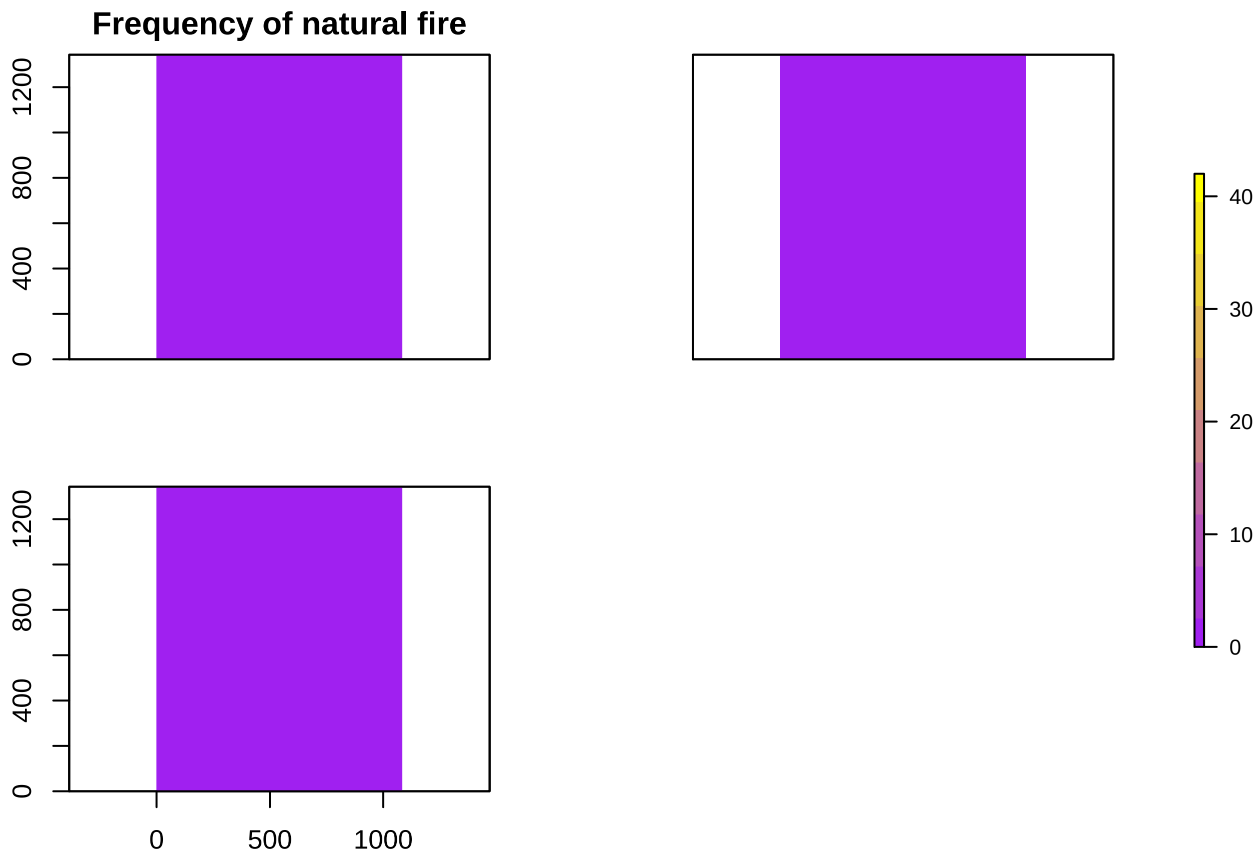

```r
VarSatck = stack(RasterStack_AllF, RasterStack_AccF, RasterStack_NatF,RasterStack_RxF)
  Max = max(maxValue(VarSatck))+1
  Min = min(minValue(VarSatck))
  brks = seq(Min,Max,by=0.1)
  nbrks = length(brks)-1
  r.range = c(Min, Max)

  rasTot = raster(ncol=2, nrow=1)
values(rasTot) = c(Min,Max)

par(mfrow=c(1,3))

plot(VarSatck, breaks=brks,col=pal.2(nbrks), legend = F, zlim=c(Min,Max),
main = "Frequency of all fire types")
plot(rasTot, legend.only=TRUE, col=pal.2(nbrks),
    legend.width=1, legend.shrink=0.75,
    legend.args=list(text="??",
                     side=4, font=2, line=2.5, cex=0.8))
```



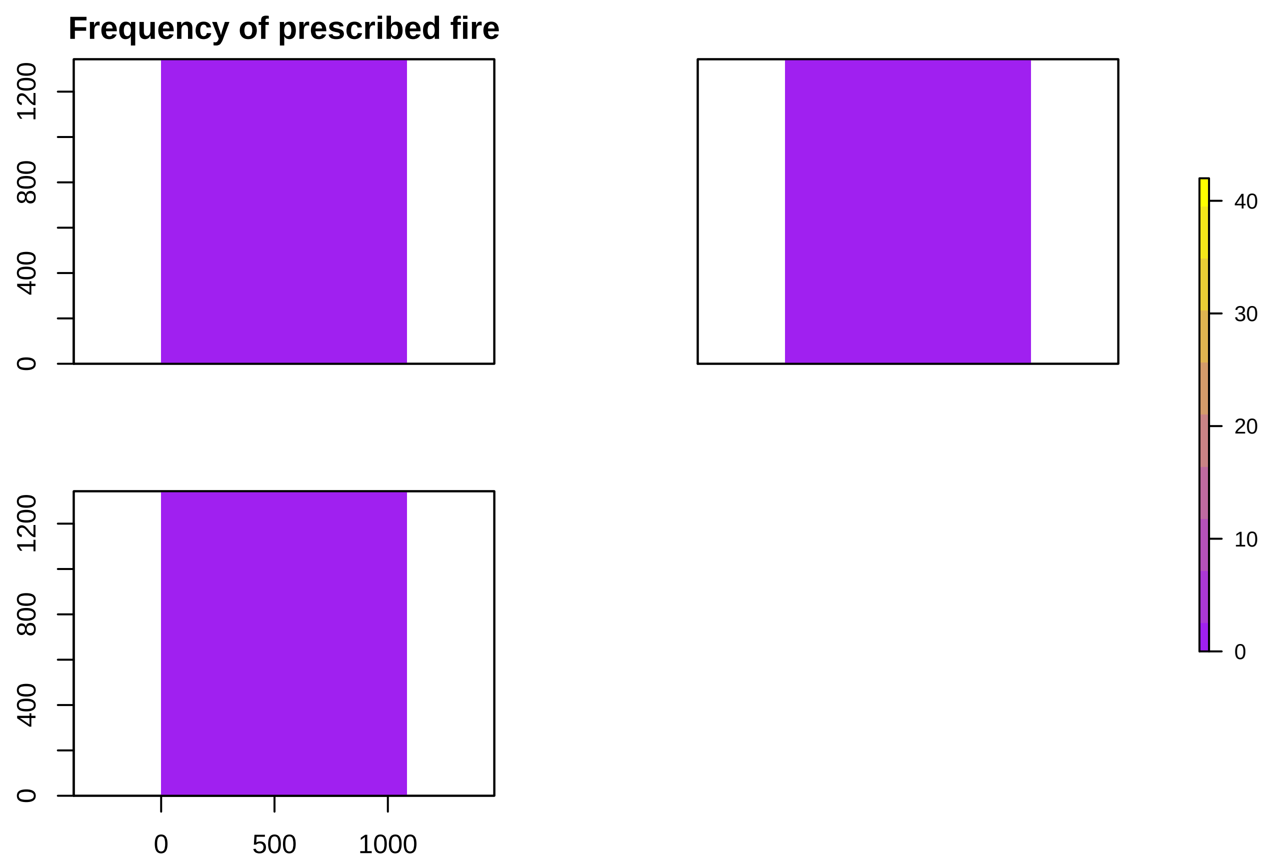

```
VarSatck = RasterStack_AccF
  Max = max(maxValue(VarSatck))+1
  Min = min(minValue(VarSatck))
  brks = seq(Min,Max,by=0.1)
  nbrks = length(brks)-1
  r.range = c(Min, Max)

par(mfrow=c(1,3))

plot(VarSatck, breaks=brks,col=pal.2(nbrks), legend = F, zlim=c(Min,Max),
main = "Frequency of accidental fire")
plot(rasTot, legend.only=TRUE, col=pal.2(nbrks),
    legend.width=1, legend.shrink=0.75,
    legend.args=list(text="??",
                     side=4, font=2, line=2.5, cex=0.8))
```




```
VarSatck = RasterStack_NatF
  Max = max(maxValue(VarSatck))+1
  Min = min(minValue(VarSatck))
  brks = seq(Min,Max,by=0.1)
  nbrks = length(brks)-1
  r.range = c(Min, Max)

par(mfrow=c(1,3))

plot(VarSatck, breaks=brks,col=pal.2(nbrks), legend = F, zlim=c(Min,Max),
main = "Frequency of natural fire")
plot(rasTot, legend.only=TRUE, col=pal.2(nbrks),
    legend.width=1, legend.shrink=0.75,
    legend.args=list(text="??",
                     side=4, font=2, line=2.5, cex=0.8))
```



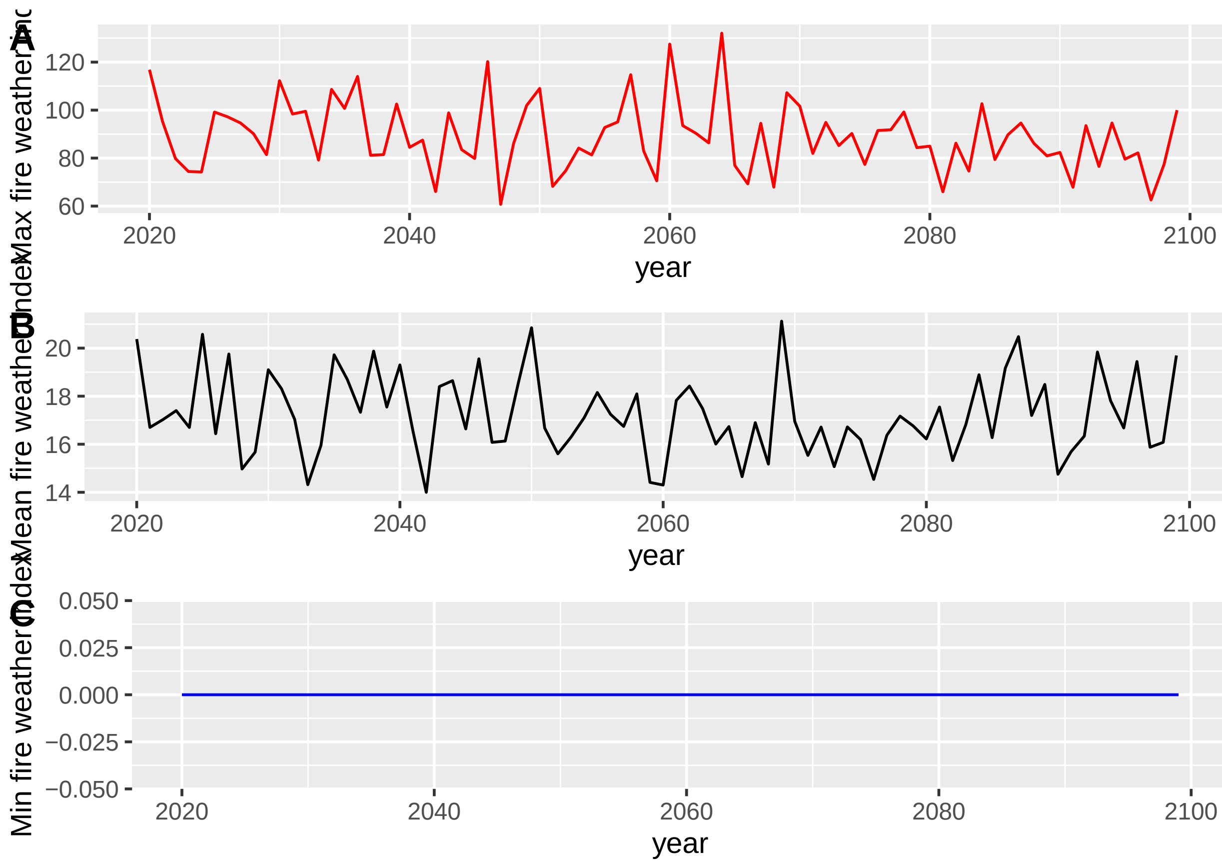

```
VarSatck = RasterStack_RxF
  Max = max(maxValue(VarSatck))+1
  Min = min(minValue(VarSatck))
  brks = seq(Min,Max,by=0.1)
  nbrks = length(brks)-1
  r.range = c(Min, Max)

par(mfrow=c(1,3))

plot(VarSatck, breaks=brks,col=pal.2(nbrks), legend = F, zlim=c(Min,Max),
main = "Frequency of prescribed fire")
plot(rasTot, legend.only=TRUE, col=pal.2(nbrks),
    legend.width=1, legend.shrink=0.75,
    legend.args=list(text="??",
                     side=4, font=2, line=2.5, cex=0.8))
```



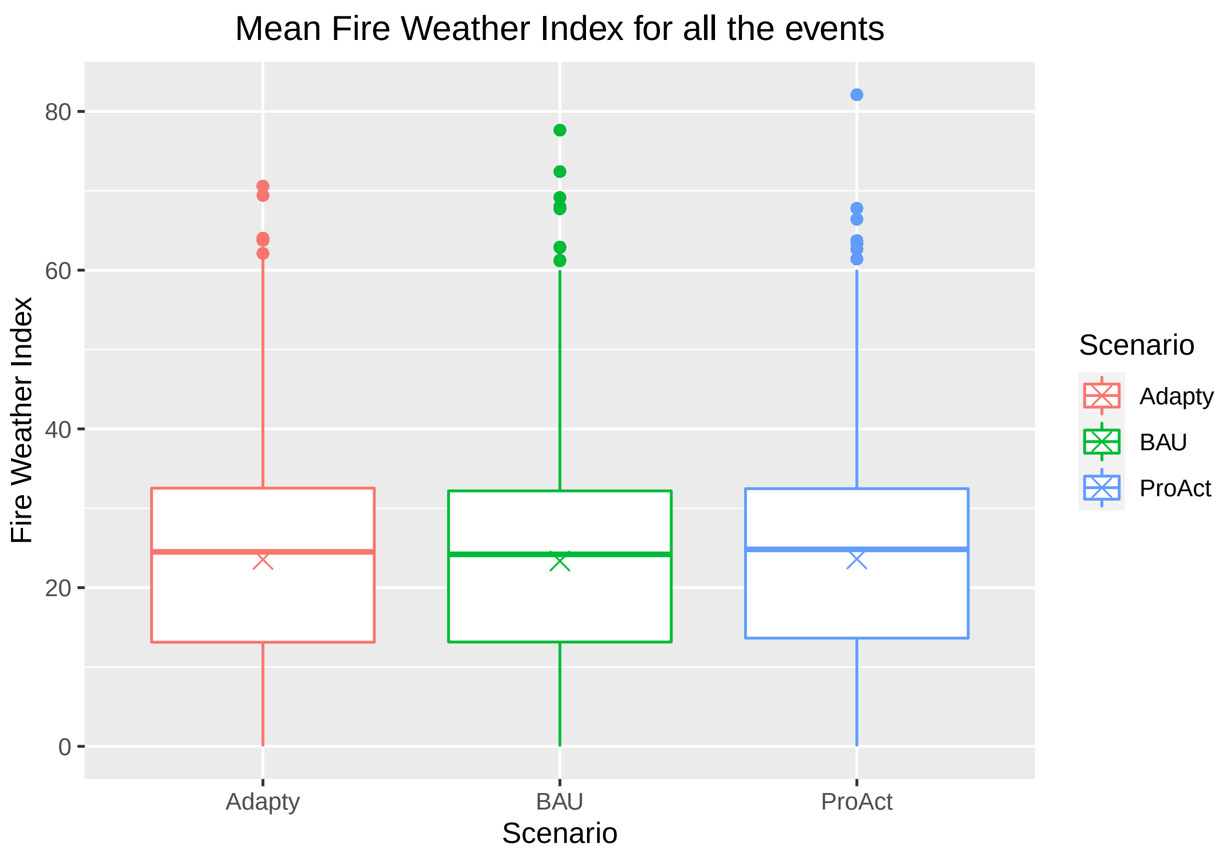

```
FWI_Tot = c("Year", "FWI", "FWI_Min","FWI_Max", "T_Min", "T_Max", "Ppt_Total","Scenario")

Event_Tot = c("Year", "Day", "FWI", "Scenario")

for(scenario in ListScenario){

    if(scenario == "BAU") {Input = BAU_input
    } else if(scenario == "Adapty"){Input = Adapty_input
    } else {Input = ProAct_input}

  Clim = paste(Input,"..1/Climate-future-input-log.csv",sep="") %>% read.csv()
  list_year = Clim$Year %>% unique()

  for(year in list_year) {
    FWI = Clim$FWI[Clim$Year == year] %>% mean()
    FWImin = Clim$FWI[Clim$Year == year] %>% min()
    FWImax = Clim$FWI[Clim$Year == year] %>% max()
    Tmin = Clim$min_airtemp[Clim$Year == year] %>% min()
    Tmax = Clim$max_airtemp[Clim$Year == year] %>% max()
    Psum = Clim$ppt[Clim$Year == year] %>% sum()

    FWI_Tot = rbind(FWI_Tot, c(year, FWI, FWImin, FWImax, Tmin, Tmax, Psum, scenario))

    Event = paste0(Input,"..1/scrapple-events-log.csv") %>% read.csv()

    EventFWI = cbind(Event$SimulationYear, Event$InitialDayOfYear, Event$MeanFWI) %>% as.data.frame()
```

```
    EventFWI$Scenario = scenario

    Event_Tot = rbind(Event_Tot, EventFWI)
  }
}

FWI_Tot = as.data.frame(FWI_Tot)
colnames(FWI_Tot)=FWI_Tot[1,] %>% unlist %>% as.character
FWI_Tot=FWI_Tot[-1,]

j=c(1:7)

FWI_Tot[ , j] = apply(FWI_Tot[ , j], 2,
                    function(x) as.numeric(as.character(x)))

Event_Tot = as.data.frame(Event_Tot)
colnames(Event_Tot)=Event_Tot[1,] %>% unlist %>% as.character
Event_Tot=Event_Tot[-1,]

j=c(1:3)

Event_Tot[ , j] = apply(Event_Tot[ , j], 2,
                    function(x) as.numeric(as.character(x)))
```

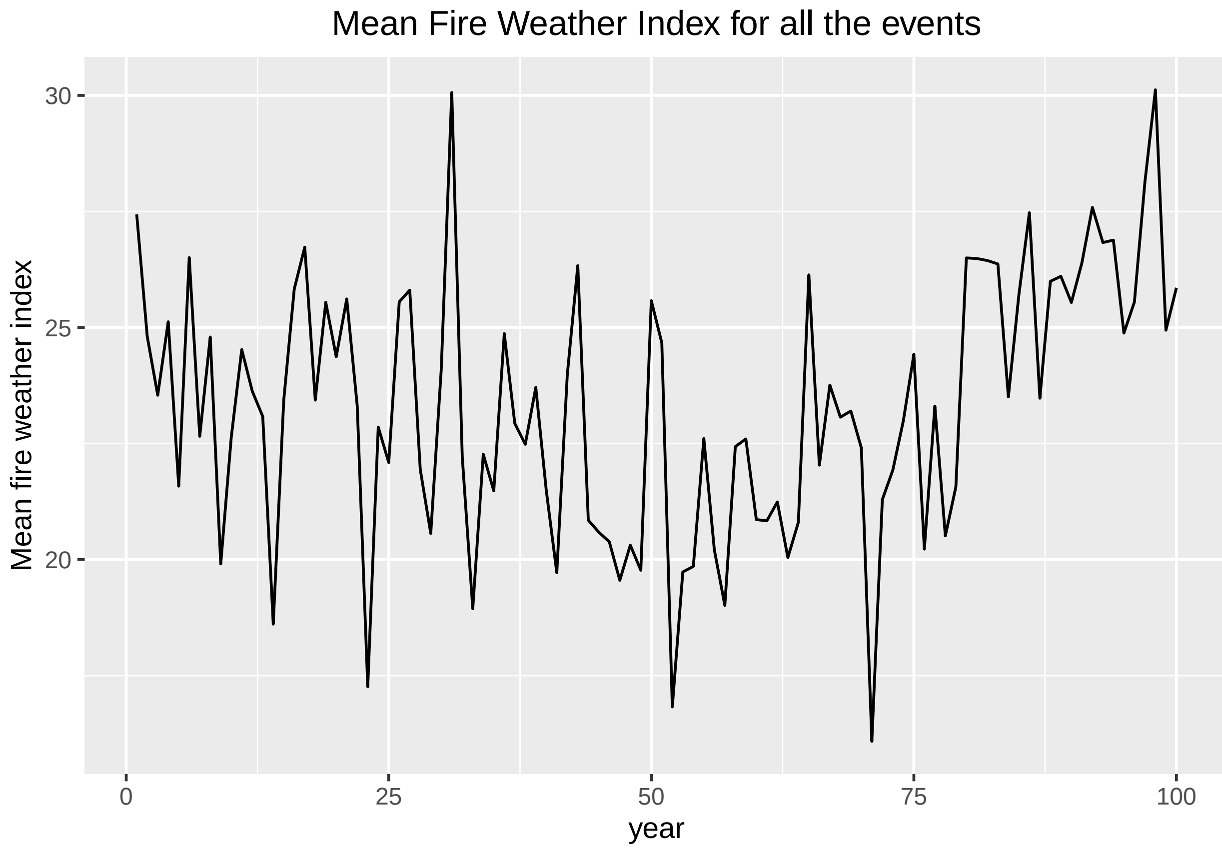
```
FWIp = ggplot(FWI_Tot, aes(x=Year, y=FWI)) +
  geom_line() +
  xlab("year")+
  ylab("Mean fire weather index")

Maxp = ggplot(FWI_Tot, aes(x=Year, y=FWI_Max)) +
  geom_line(color="red") +
  xlab("year")+
  ylab("Max fire weather index")

Minp = ggplot(FWI_Tot, aes(x=Year, y=FWI_Min)) +
  geom_line(color="blue") +
  xlab("year")+
  ylab("Min fire weather index")

plot_grid(Maxp, FWIp, Minp, labels="AUTO", ncol = 1, nrow = 3)
```



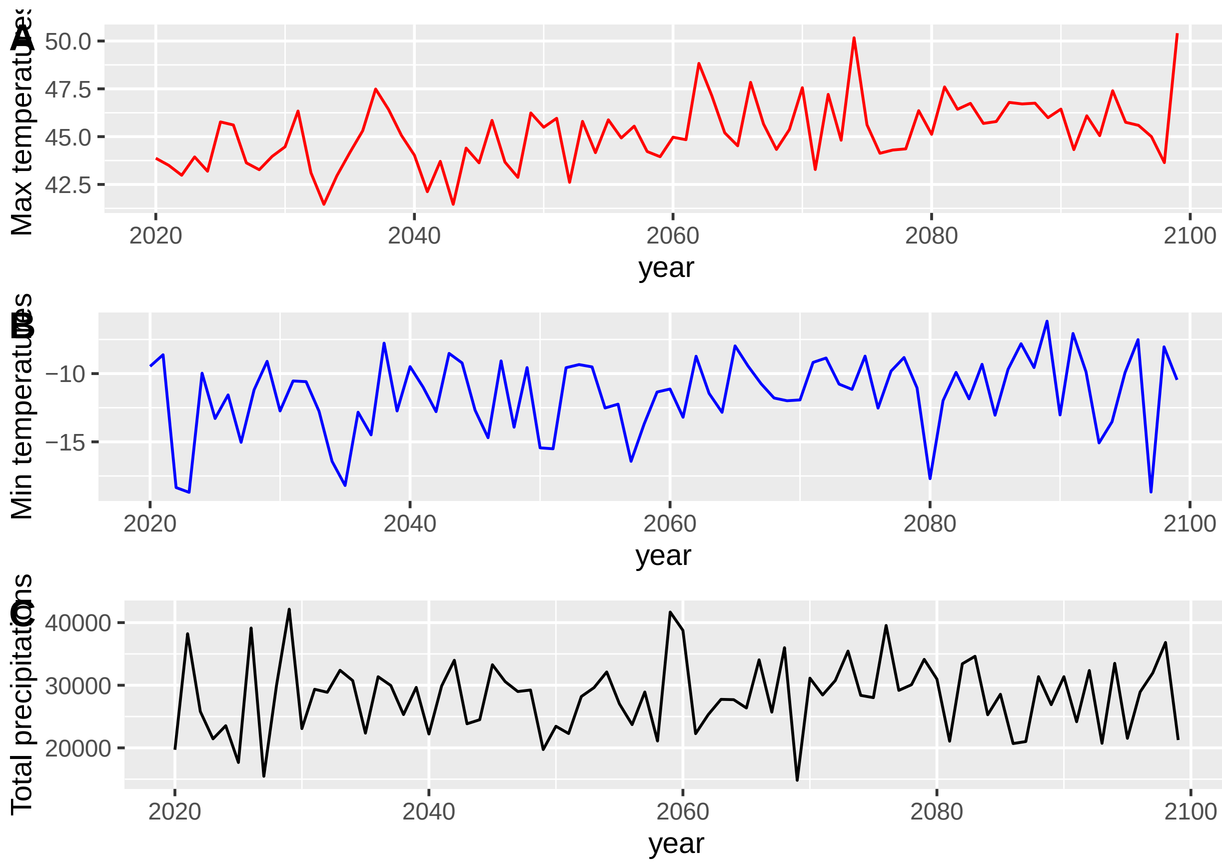

```r
ggplot(Event_Tot, aes(x=Scenario, y=FWI, col=Scenario)) +
  geom_boxplot() +
  stat_summary(fun=mean, geom="point", shape=4, size=3) +
  ylab("Fire Weather Index") +
  ggtitle("Mean Fire Weather Index for all the events") +
  theme(plot.title = element_text(hjust = 0.5))
```




```r
Event_Ag = aggregate(data=Event_Tot, FWI~Year, mean)

ggplot(Event_Ag, aes(x=Year, y=FWI)) +
  geom_line() +
  xlab("year")+
  ylab("Mean fire weather index")+
  ggtitle("Mean Fire Weather Index for all the events") +
  theme(plot.title = element_text(hjust = 0.5))
```




```
Tminp = ggplot(FWI_Tot, aes(x=Year, y=T_Min)) +
  geom_line(color="blue") +
  xlab("year")+
  ylab("Min temperatures")

Tmaxp = ggplot(FWI_Tot, aes(x=Year, y=T_Max)) +
  geom_line(color="red") +
  xlab("year")+
  ylab("Max temperatures")

Pp = ggplot(FWI_Tot, aes(x=Year, y=Ppt_Total)) +
  geom_line() +
  xlab("year")+
  ylab("Total precipitations")

plot_grid(Tmaxp, Tminp, Pp, labels="AUTO", ncol = 1, nrow = 3)
```



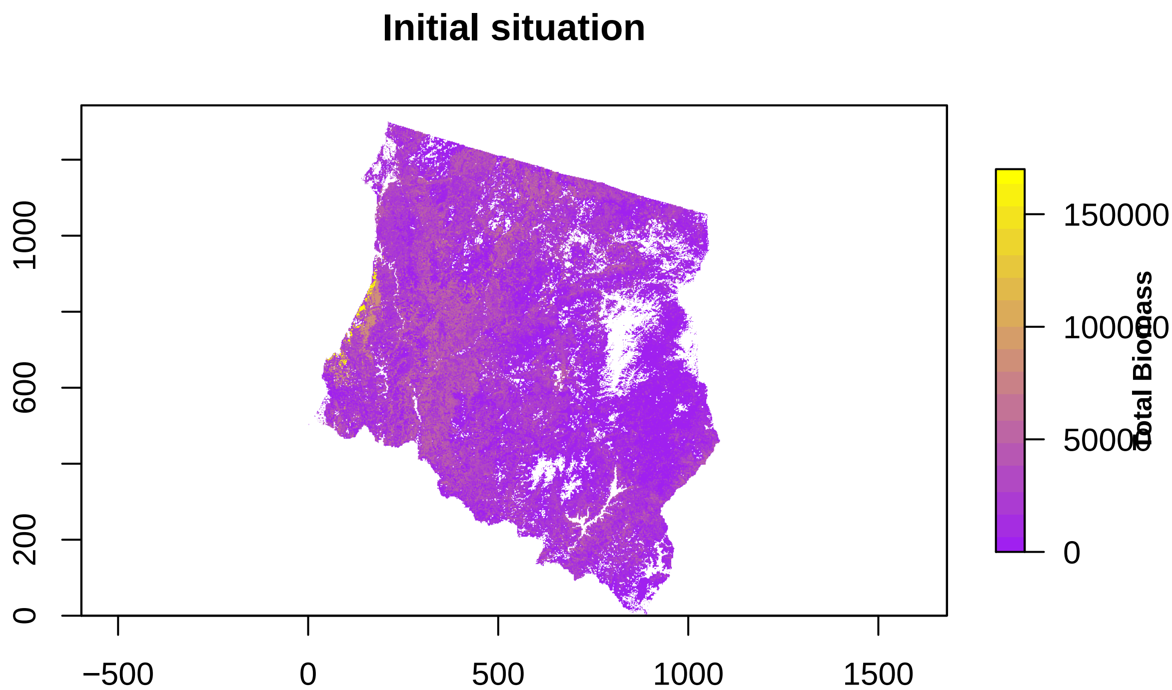
VI. Forest sustainability and biodiverity

VI.a) Sustaiability of the forest

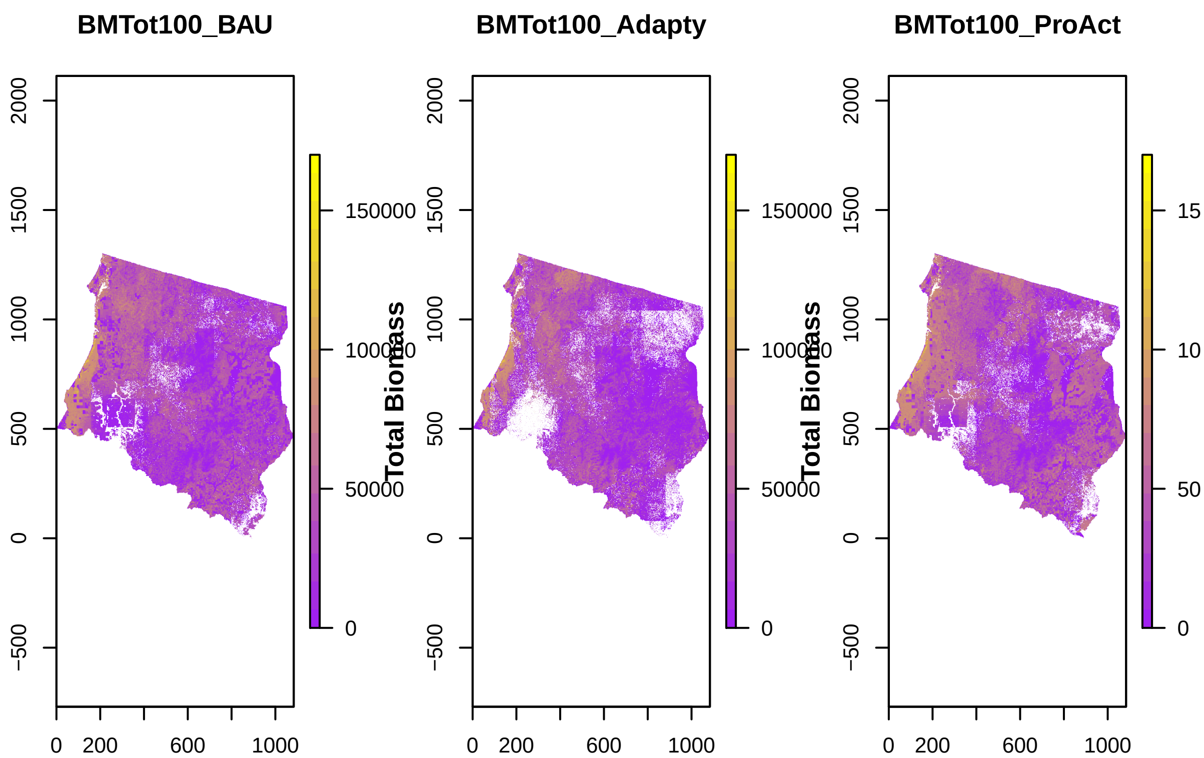
VI.a.1) Repartition of the biomass

```
StackCovDif = stack()
StackBM = stack()

for(scenario in ListScenario){

  if(scenario == "BAU") {Input = BAU_inputBM
  } else if(scenario == "Adapty"){Input = Adapty_inputBM
  } else {Input = ProAct_inputBM}

  FxR0 = paste0(Input,"FX_R_SEED-0.tif") %>% raster()
  FxR100 = paste0(Input,"FX_R_SEED-100.tif") %>% raster()

  NFxR0 = paste0(Input,"NOFX_R_SEED-0.tif") %>% raster()
  NFxR100 = paste0(Input,"NoFX_R_SEED-100.tif") %>% raster()

  NFxNR0 = paste0(Input,"NOFX_NOR_SEED-0.tif") %>% raster()
  NFxNR100 = paste0(Input,"NoFX_NOR_SEED-100.tif") %>% raster()
```

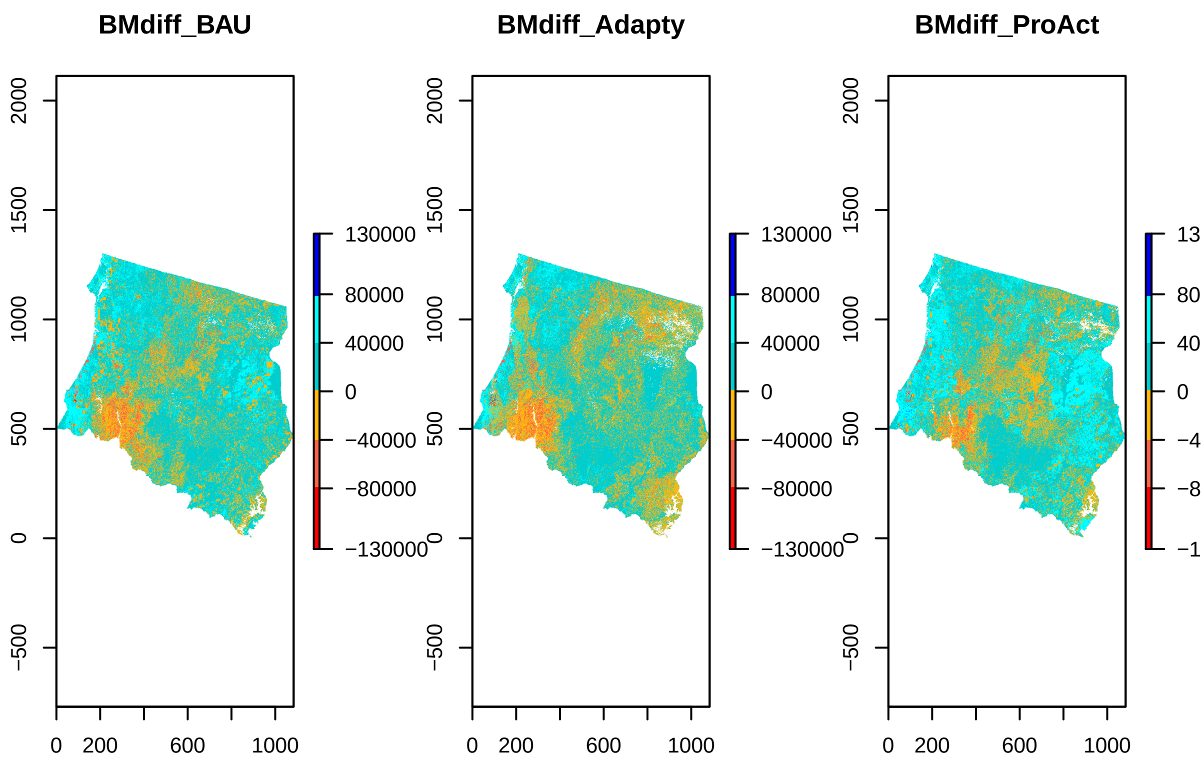
```r
  Shrub0 = FxR0 + NFxR0 + NFxNR0
  Shrub100 = FxR100 + NFxR100 + NFxNR100

  BMTot0 = paste0(Input,"TotalBiomass-0.tif") %>% raster()
  BMTot100 = paste0(Input,"TotalBiomass-100.tif") %>% raster()

  BMTot0[BMTot0 >= 300000] = NA

  BMdiff = BMTot100-BMTot0

  BMTot0[BMTot0 == 0] = NA
  BMTot100[BMTot100 == 0] = NA
  BMdiff[BMdiff == 0] = NA

  Covert0 = 1-(Shrub0/BMTot0)
  Covert100 = 1-(Shrub100/BMTot100)

  CovDif = Covert100-Covert0

  names(CovDif) = paste0("CovDif_",scenario)
  StackCovDif = addLayer(StackCovDif, CovDif)

  names(BMTot0) = paste0("BMTot0_",scenario)
  names(BMTot100) = paste0("BMTot100_",scenario)
  names(BMdiff) = paste0("BMdiff_",scenario)
  StackBM = addLayer(StackBM, BMTot0,BMTot100,BMdiff)

}

Title = "Total Biomass"
Min = 0
Max = 170000
brks = seq(Min,Max,by=10000)
nbrks = length(brks)-1
r.range = c(Min, Max)

rasTot = raster(ncol=2, nrow=1)
values(rasTot) = c(Min,Max)

tmp = StackBM[[1]]
plot(tmp, breaks=brks,col=pal.2(nbrks), legend = F, zlim=c(Min,Max),
main = "Initial situation")
plot(rasTot, legend.only=TRUE, col=pal.2(nbrks),
    legend.width=1, legend.shrink=0.75,
    legend.args=list(text=Title, side=4, font=2, line=2.5, cex=0.8))
```




```
par(mfrow=c(1,3))
for(i in c(2,5,8)){
  tmp = StackBM[[i]]
  plot(tmp, breaks=brks,col=pal.2(nbrks), legend = F, zlim=c(Min,Max),
  main = names(tmp))
  plot(rasTot, legend.only=TRUE, col=pal.2(nbrks),
      legend.width=1, legend.shrink=0.75,
      legend.args=list(text=Title, side=4, font=2, line=2.5, cex=0.8))
}
```



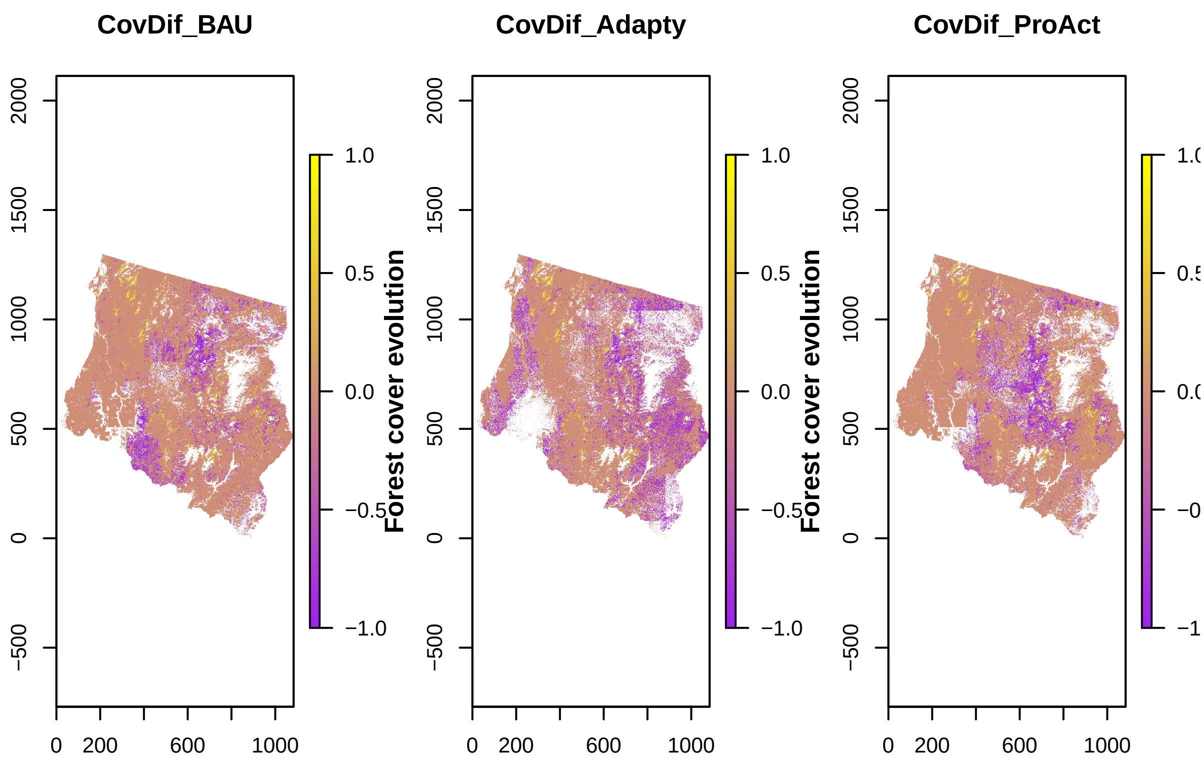

```
Title = "Total Biomass"
Min = -130000
Max = 130000
brks = c(-130000, -80000,-40000, 0, 40000, 80000,130000)
nbrks = length(brks)-1
r.range = c(Min, Max)

rasTot = raster(ncol=2, nrow=1)
values(rasTot) = c(Min,Max)

par(mfrow=c(1,3))
for(i in c(3,6,9)){
  tmp = StackBM[[i]]
  plot(tmp, breaks=brks,
       col=pal.dif(6), legend = T, zlim=c(Min,Max),
  main = names(tmp))
  }
```



VI.a.2) Patterns of regeneration

VI.a.3) Evolution of the forest cover

```
Title = "Forest cover evolution"
Min = min(minValue(StackCovDif))
Max = max(maxValue(StackCovDif))
brks = seq(Min,Max,by=0.001)
nbrks = length(brks)
r.range = c(Min, Max)

rasTot = raster(ncol=2, nrow=1)
values(rasTot) = c(Min,Max)

par(mfrow=c(1,3))
for(i in c(1:3)){
  tmp = StackCovDif[[i]]
  plot(tmp, breaks=brks,col=pal.2(nbrks), legend = F, zlim=c(Min,Max),
  main = names(tmp))
  plot(rasTot, legend.only=TRUE, col=pal.2(nbrks),
      legend.width=1, legend.shrink=0.75,
      legend.args=list(text=Title, side=4, font=2, line=2.5, cex=0.8))
}
```



IV.b) Biodiverity on the forest

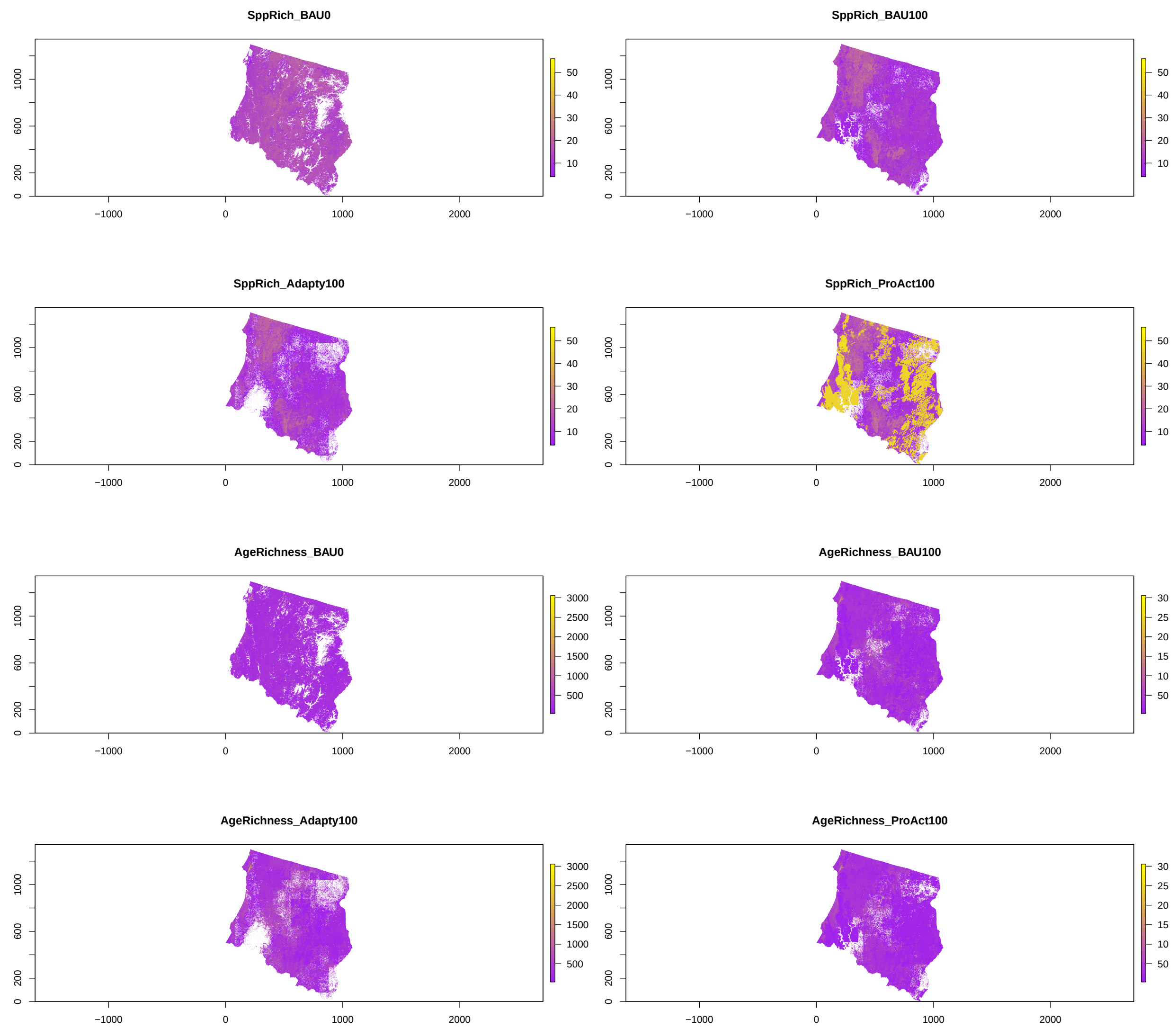
Don't take into account the 0.

```
Biodiv = c(c("Year", "Scenario", "Spp_Rich_mean", "Age_Richness_mean"))

Stack=stack()

for(year in seq(0, 100, by=10)){
  for(scenario in ListScenario){

    if(scenario == "BAU") {Input = BAU_input
    } else if(scenario == "Adapty"){Input = Adapty_input
    } else {Input = ProAct_input}

    SppRich = paste(Input,"..1/outputs/spp-counts/SPP-RICH-",
                    year,".img",sep="") %>% raster()
    SppRich[SppRich == 0] = NA
    SppRich_mean = cellStats(SppRich, stat='mean', na.rm=TRUE)

    if(year %in% c(0, 100)){
      Q = quantile(SppRich, probs = c(0.25, 0.75))
      SppRichQ = SppRich
      SppRichQ[SppRichQ <= Q[2] & SppRichQ > Q[1]] = NA
      SppRichQ[SppRichQ <= Q[1]] = 0
```

```
      SppRichQ[SppRichQ > Q[2]] = 1

      names(SppRich) = paste0("SppRich_",scenario,year)
      names(SppRichQ) = paste0("SppRichQ_",scenario,year)
      Stack = addLayer(Stack, SppRich, SppRichQ)
    }
    AgeRichness = paste(Input,"..1/outputs/age-all-spp/AGE-AVG-",
                        year,".img",sep="") %>% raster()
    AgeRichness[AgeRichness == 0] = NA
    AgeRichness_mean = cellStats(AgeRichness, stat='mean', na.rm=TRUE)

    if(year%in% c(0, 100)){
      Q = quantile(AgeRichness, probs = c(0.25, 0.75))
      AgeRichnessQ = AgeRichness
      AgeRichnessQ[AgeRichnessQ <= Q[2] & AgeRichnessQ > Q[1]] = NA
      AgeRichnessQ[AgeRichnessQ <= Q[1]] = 0
      AgeRichnessQ[AgeRichnessQ > Q[2]] = 1

      names(AgeRichness) = paste0("AgeRichness_",scenario,year)
      names(AgeRichnessQ) = paste0("AgeRichnessQ_",scenario,year)
      Stack = addLayer(Stack, AgeRichness, AgeRichnessQ)
    }

    Biodiv = rbind(Biodiv, c(year, scenario, SppRich_mean, AgeRichness_mean))
    }
  }

colnames(Biodiv)=Biodiv[1,] %>% unlist %>% as.character
Biodiv=Biodiv[-1,] %>% as.data.frame()

j=c(1,3,4)

Biodiv[ , j] = apply(Biodiv[ , j], 2,
                     function(x) as.numeric(as.character(x)))
```

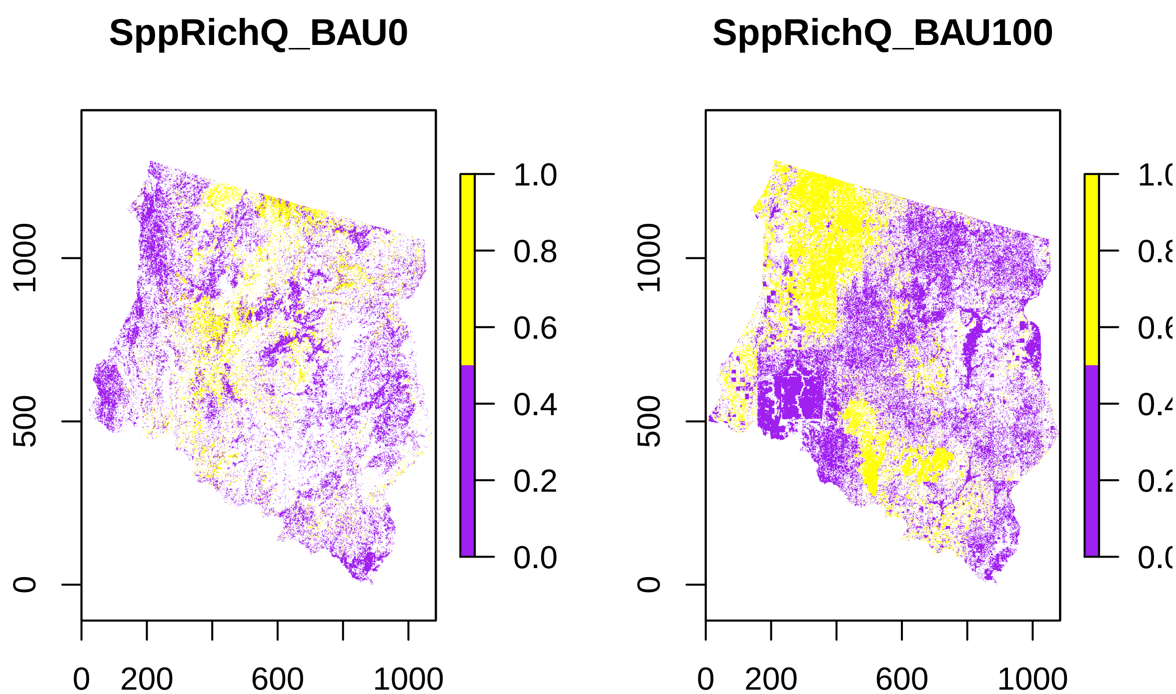
```
# par(mar=c(4.1,4.1,12.4,10.5))

for(i in c(1,3)){
  VarSatck = stack(Stack[[i]],Stack[[i+12]],
                   Stack[[i+16]],Stack[[i+20]])
  Max = max(maxValue(VarSatck))+1
  Min = min(minValue(VarSatck))
  brks = seq(Min,Max,by=0.1)
  nbrks = length(brks)-1
  r.range = c(Min, Max)

  rasTot = sum(VarSatck)

  par(mfrow=c(1,2))
  for(i in seq_len(nlayers(VarSatck))){
    tmp = VarSatck[[i]]
    plot(tmp, breaks=brks,col=pal.2(nbrks), legend = F, zlim=c(Min,Max),
    main = names(tmp))
```

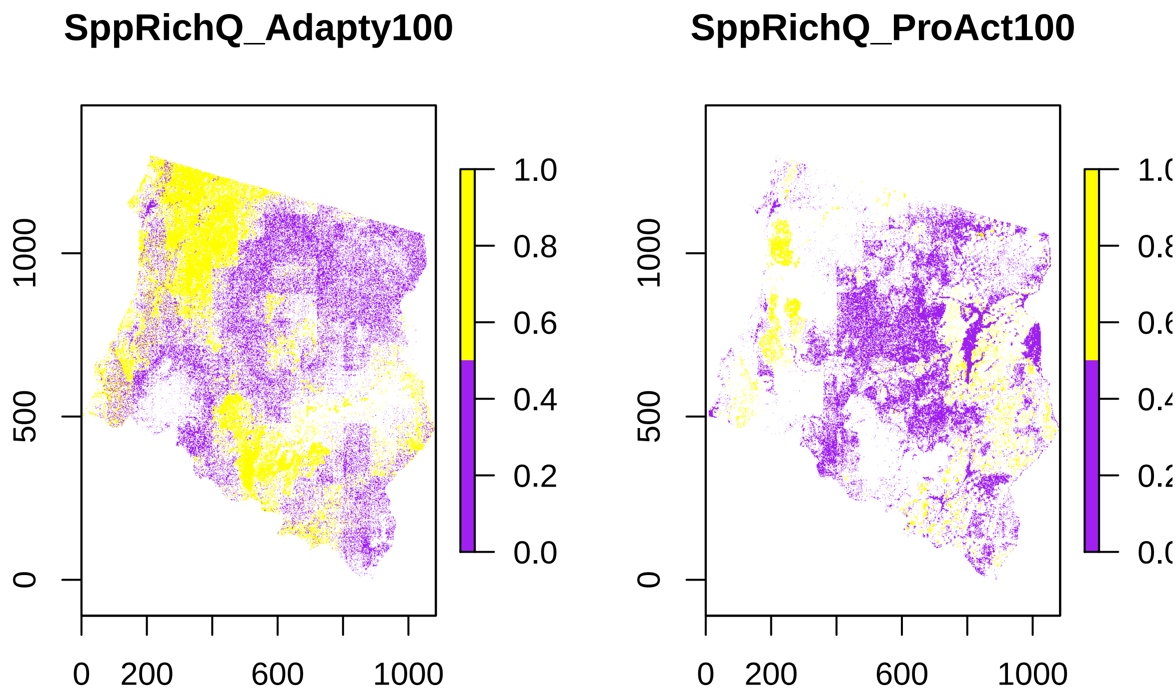
```
    plot(rasTot, legend.only=TRUE, col=pal.2(nbrks),
       legend.width=1, legend.shrink=0.75)
  }}
```

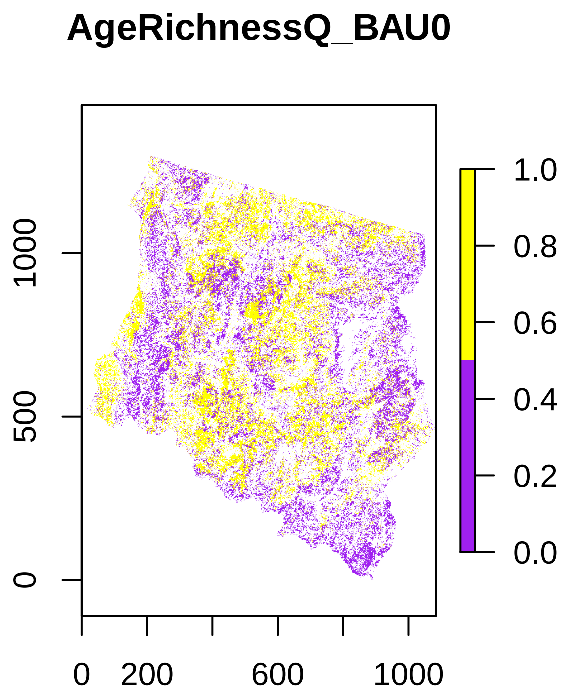


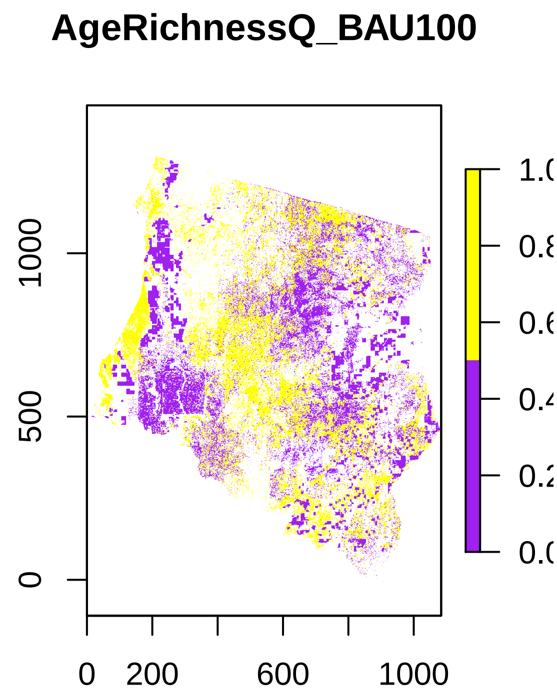

```
for(i in c(2,4)){
  VarSatck = stack(Stack[[i]],Stack[[i+12]],
                   Stack[[i+16]],Stack[[i+20]])
  Max = 1
  Min = 0
  brks = c(0, 0.5, 1)
  nbrks = 2
  r.range = c(Min, Max)

  rasTot = raster(ncol=2, nrow=1)
  values(rasTot) = c(Min,Max)

  par(mfrow=c(1,2))
```

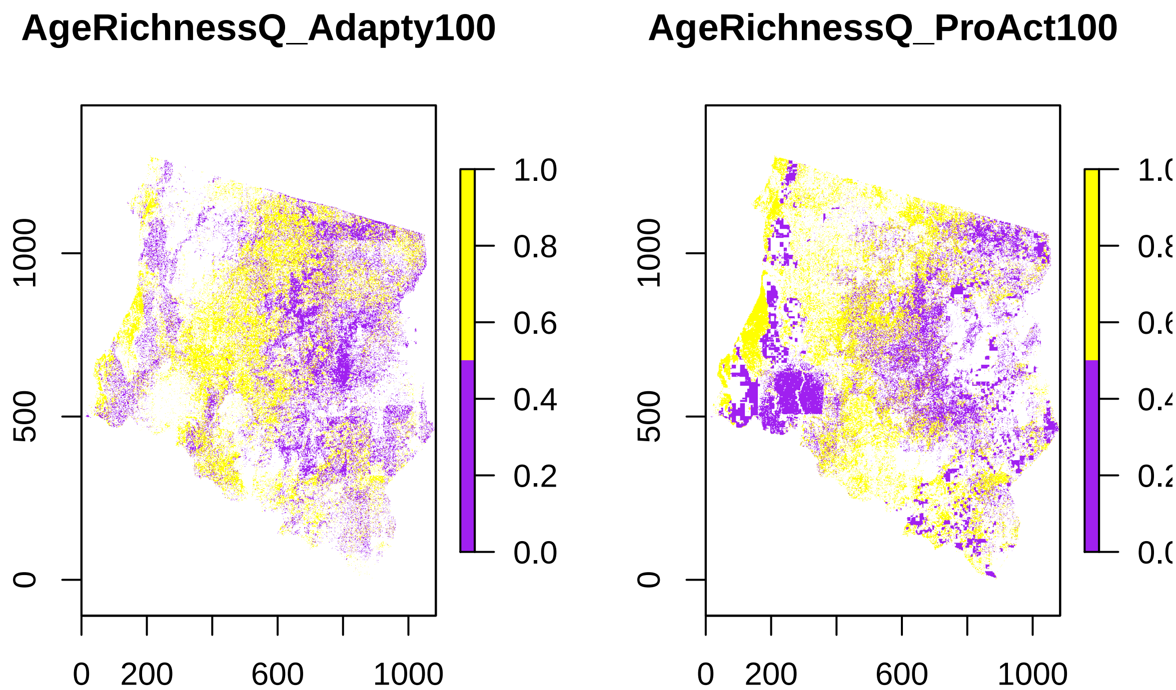
```r
  for(i in seq_len(nlayers(VarSatck))){
    tmp = VarSatck[[i]]
    plot(tmp, breaks=brks,col=pal.2(nbrks), legend = F, zlim=c(Min,Max),
    main = names(tmp))
    plot(rasTot, legend.only=TRUE, col=pal.2(nbrks),
       legend.width=1, legend.shrink=0.75)
}}
```



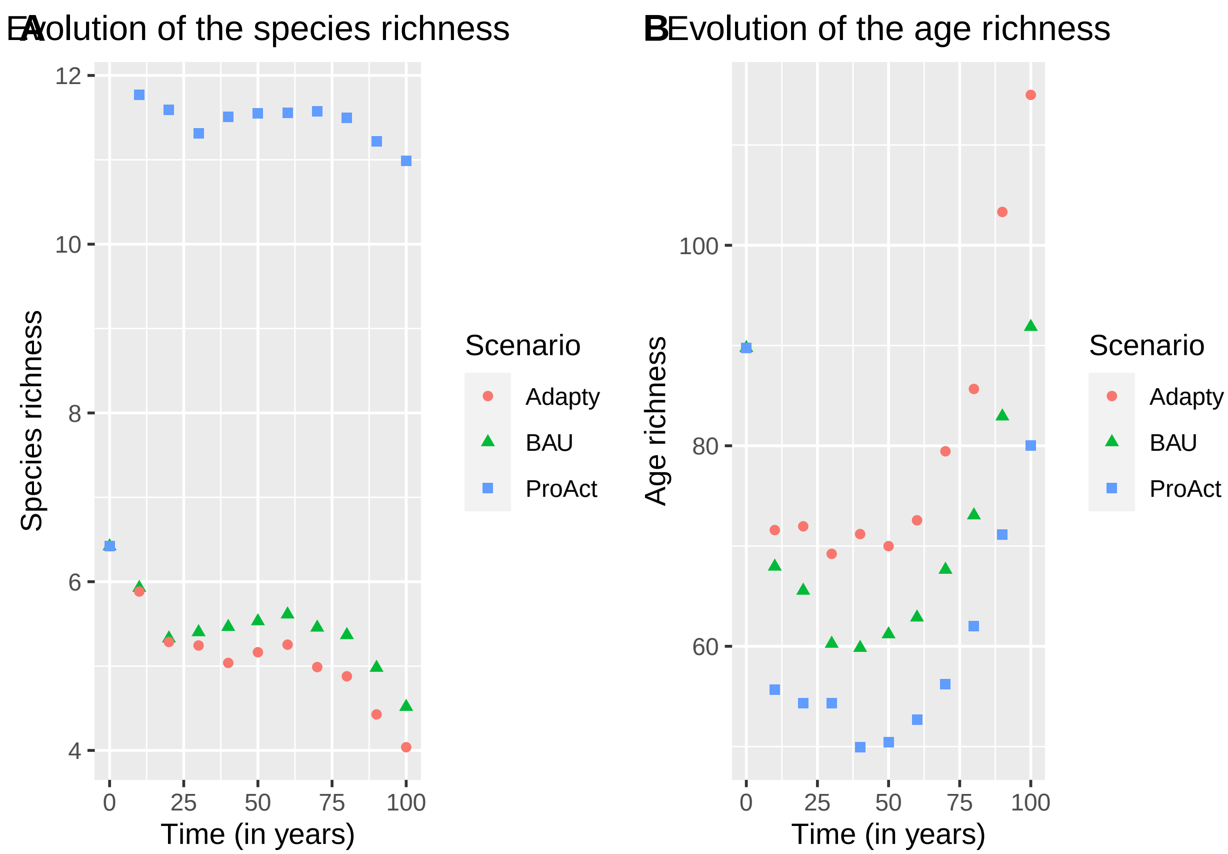

SppRichQ_Adapty100
SppRichQ_ProAct100
0
500
1000
0
200
600
1000
0.0
0.2
0.4
0.6
0.8
1.0


AgeRichnessQ_BAU0
1000
500
0
0
200
600
1000
1.0
0.8
0.6
0.4
0.2
0.0


AgeRichnessQ_BAU100
1000
500
0
0
200
600
1000




```r
pSR = ggplot(Biodiv, aes(x=Year, y=Spp_Rich_mean, col=Scenario, shape = Scenario)) +
  geom_point() +
  scale_x_continuous(name= "Time (in years)") +
  scale_y_continuous(name = "Species richness")+#, limits =c(2.126, 2.127))
  ggtitle("Evolution of the species richness") +
  theme(plot.title = element_text(hjust = 0.5))

pAR = ggplot(Biodiv, aes(x=Year, y=Age_Richness_mean, col=Scenario, shape = Scenario)) +
  geom_point() +
  scale_x_continuous(name= "Time (in years)") +
  scale_y_continuous(name = "Age richness")+#, limits =c(2.126, 2.127))
  ggtitle("Evolution of the age richness") +
  theme(plot.title = element_text(hjust = 0.5))


  plot_grid(pSR, pAR, labels="AUTO", ncol = 2, nrow = 1)
```


A
Evolution of the species richness
Species richness
12
10
8
6
4
0
25
50
75
100
Time (in years)
Scenario
Adapty
BAU
ProAct
B
Evolution of the age richness
Age richness
100
80
60
0
25
50
75
100
Time (in years)
Scenario
Adapty
BAU
ProAct