

# Are environmental and geographic effective surrogates for genetic variation in conservation planning?

Jeffrey O. Hanson<sup>1</sup>, Jonathan R. Rhodes<sup>2</sup>, Cynthia Riginos<sup>2</sup>, Hugh P. Possingham<sup>1</sup>,  
Richard A. Fuller<sup>1</sup>

<sup>1</sup>*School of Biological Sciences, The University of Queensland, Brisbane, QLD, Australia*

<sup>2</sup>*School of Geography, Planning and Environmental Management, The University of  
Queensland, Brisbane, QLD, Australia*

Correspondance should be addressed to [jeffrey.hanson@uqconnect.edu.au](mailto:jeffrey.hanson@uqconnect.edu.au)

08 December 2015

## Abstract

Insert abstract here.

## Contents

<b>Introduction</b>	1
<b>Methods</b>	1
Study area	1
Genomic data	3
Surrogate data	5
Prioritisations	6
<b>Results</b>	22
Single species prioritisations	22
Multi-species prioritisations	23
Pareo-frontier analysis	26
<b>Discussion</b>	27
<b>Acknowledgements</b>	27
<b>References</b>	27
<b>Supporting Information</b>	27
Appendix S1: Species distributions	27
Appendix S2: BayeScan Results	31
Appendix S3: Genomic MDS	44
Appendix S4: Distribution maps of intra-specific variation	45
Appendix S5: Principle components analysis on climatic variation	49
Appendix S6: Maps of climatic variation	50

## Introduction

## Methods

### Study area

To address the aims of this study, we obtained species distribution and genomic (AFLP) from (Meirmans *et al.* 2011). This dataset was chosen because it provides genomic data for a multitude

of species at a high spatial resolution (approx.  $20\text{km}^2 \times 22\text{km}^2$ ).

```
## compile spatial grid data
# load grid cell centroids
grid.DF <- fread(
  'extdata/Data_Meirmans_et_al_IntrabioDiv/ReadMe.txt',
  data.table=FALSE,
  skip='cell\tLong\tLat'
) %>% rename(
  grid.longitude=Long,
  grid.latitude=Lat
) %>% mutate(
  id=seq_along(grid.latitude)
)

# load in aflu data
spp.aflu.paths <- dir(
  'extdata/Data_Meirmans_et_al_IntrabioDiv',
  '^.*AFLP\\.dat$',
  full.names=TRUE
)[seq_len(n.spp)]
spp.BayeScanData.LST <- llply(
  spp.aflu.paths,
  read.BayeScanData
)

## compile species occurrence data
# load in data
spp.loc.paths <- dir(
  'extdata/Data_Meirmans_et_al_IntrabioDiv',
  '^.*locations\\.txt$',
  full.names=TRUE
)[seq_len(n.spp)]
spp.samples.DF <- ldply(
  seq_along(spp.loc.paths),
  .fun=function(i) {
    x <- mutate(
      fread(spp.loc.paths[i], data.table=FALSE),
      species=gsub('_locations.txt', '', basename(spp.loc.paths[i]), fixed=TRUE)
    ) %>% rename(
      cell=population,
      sample.longitude=longitude,
      sample.latitude=latitude
    )
    return(x[as.numeric(spp.BayeScanData.LST[[i]]@populations),])
  }
) %>% left_join(
  grid.DF,
```

```

        by='cell'
    )
    # append species data to grid data.frame (wide-format)
    for (i in unique(spp.samples.DF$species))
        grid.DF[[i]] <- replace(
            rep(0, nrow(grid.DF)),
            which(grid.DF$cell %in% filter(spp.samples.DF, species==i)$cell),
            1
        )

```

## Genomic data

Loci in the AFLP were classified as adaptive or neutral using BayeScan (version 2.1) using a probability threshold of 0.5).

```

# assign cells as populations
spp.BayeScanData.LST <- llply(
    seq_along(unique(spp.samples.DF$species)),
    function(i) {
        bd <- spp.BayeScanData.LST[[i]]
        bd@populations <- filter(spp.samples.DF, species==unique(spp.samples.DF$species)[i])$cell
        return(bd)
    }
)

# run BayeScan
spp.BayeScan.LST <- llply(
    spp.BayeScanData.LST,
    run.BayeScan,
    threshold=bs.threshold,
    threads=bs.threads,
    n=bs.n,
    thin=bs.thin,
    nbp=bs.nbp,
    pilot=bs.pilot,
    burn=bs.burn
)

# run MDS
spp.mds.LST <- llply(
    spp.BayeScan.LST,
    function(i) {
        names<-c('adaptive', 'neutral'), function(j) {
            if (sum(i@results@fst==j)==0)
                return(NULL)
            return(
                mds(
                    i,
                    metric='gower',

```

```

        type=j,
        k=mds.k,
        trymax=mds.trymax
    )
}
}), c('adaptive', 'neutral'))
}
)

```

```

## Warning in daisy(cbind(as.data.frame(x@matrix == 1), 1), metric = metric, :
## at least one binary variable has not 2 different levels.

```

```

## Run 0 stress 0.2387515
## Run 1 stress 0.2393767
## Run 2 stress 0.2441548
## Run 0 stress 0.2249186
## Run 1 stress 0.225666
## Run 2 stress 0.2259804

```

```

## Warning in daisy(cbind(as.data.frame(x@matrix == 1), 1), metric = metric, :
## at least one binary variable has not 2 different levels.

```

```

## Run 0 stress 0.25157
## Run 1 stress 0.2575878
## Run 2 stress 0.2675034

```

```

## Warning in daisy(cbind(as.data.frame(x@matrix == 1), 1), metric = metric, :
## at least one binary variable has not 2 different levels.

```

```

## Run 0 stress 0.2127055
## Run 1 stress 0.2177005
## Run 2 stress 0.2123847
## ... New best solution
## ... procrustes: rmse 0.03672649  max resid 0.1819873
## Run 0 stress 0.1957005
## Run 1 stress 0.19997
## Run 2 stress 0.2048544
## Run 0 stress 0.2316074
## Run 1 stress 0.241015
## Run 2 stress 0.2355921

```

```

# store mds rotations for each sample
spp.samples.DF <- ldply(seq_along(unique(spp.samples.DF$species)), .fun=function(i) {
  x <- filter(spp.samples.DF, species==unique(spp.samples.DF$species)[i])
  for (j in c('adaptive', 'neutral')) {

```

```

    if (!is.null(spp.mds.LST[[i]][[j]])) {
      x <- cbind(
        x,
        'names<-'(
          as.data.frame(spp.mds.LST[[i]][[j]]$points),
          paste0(j, '_d', seq_len(mds.k))
        )
      )
    }
  }
  return(x)
})
# store mds average rotation for each grid
for (i in seq_along(unique(spp.samples.DF$species))) {
  for (j in c('adaptive', 'neutral')) {
    if (!is.null(spp.mds.LST[[i]][[j]])) {
      curr.sub <- filter(spp.samples.DF, species==unique(spp.samples.DF$species)[i])
      for (k in seq_len(mds.k)) {
        curr.vals <- tapply(
          curr.sub[[paste0(j, '_d', k)]],
          curr.sub$cell,
          FUN=mean
        )
        curr.pos <- match(names(curr.vals), grid.DF$cell)
        grid.DF[curr.pos, paste0(unique(spp.samples.DF$species)[i], '_', j, '_d', k)] <- curr.vals
      }
    }
  }
}

```

## Surrogate data

```

## create spatial data
# grid data as SpatialPolygonsDataFrame
grid.PTS <- SpatialPoints(as.matrix(grid.DF[,2:3]))
grid.PLY <- grid.PTS %>%
  points2grid(tolerance=0.05) %>%
  as('SpatialPolygons')
grid.PLY <- grid.PLY[sapply(gIntersects(grid.PTS, grid.PLY, byid=TRUE, returnDense=FALSE), '[0,1]')]
  spChFIDs(
    as.character(seq_len(nrow(grid.DF)))
  ) %>%
  SpatialPolygonsDataFrame(
    data=grid.DF
  )
grid.PLY@proj4string <- wgs1984

```

```

grid.PPLY <- spTransform(grid.PLY, europeEA)
# sample data as SpatialPoints
spp.sample.PTS <- SpatialPointsDataFrame(
  coords=as.matrix(spp.samples.DF[,5:6]),
  data=spp.samples.DF,
  proj4string=wgs1984
)
spp.sample.PPTS <- spTransform(spp.sample.PTS, europeEA)
## extract geographic data
centroids.DF <- gCentroid(grid.PPLY, byid=TRUE) %>% slot('coords') %>%
  as.data.frame() %>% 'names<-'(paste0('geo_d',1:2))
grid.DF <- cbind(grid.DF, centroids.DF)
## extract climatic data
# load climatic data
bioclim.STK <- stack('extdata/BioClim_variables/bioclim_pca.tif')
# extract mean for each cell for each principle component
extract.DF <- grid.PPLY %>% rasterize(bioclim.STK, field='id') %>%
  zonal(x=bioclim.STK) %>% as.data.frame() %>% select(-1) %>%
  'names<-'(paste0('env_d',seq_len(nlayers(bioclim.STK))))
# merge with grid.DF
grid.DF <- cbind(grid.DF, extract.DF)
## update spatial objects
grid.PLY@data <- grid.DF
grid.PPLY@data <- grid.DF

```

## Prioritisations

```

# generate attribute spaces for geographic and environmental data
surrogate.ASL <- llply(
  list(grep('^env.*$', names(grid.DF)),grep('^geo.*$', names(grid.DF))),
  .fun=function(x) {
    make.multi.species.AttributeSpace(
      site.data=grid.DF[,x,drop=FALSE],
      species.data=grid.DF[,unique(spp.samples.DF$species),drop=FALSE]
    )
  }
)
# generate attribute spaces for genetic data
adaptive.ASL <- llply(
  seq_along(unique(spp.samples.DF$species)),
  function(i) {
    make.single.species.AttributeSpace(
      site.data=select(grid.DF, contains(paste0(unique(spp.samples.DF$species)[i], '_adap')),
      species.data=select(filter(spp.samples.DF, species==unique(spp.samples.DF$species)[i]),
      spp.pos=i,

```

```

        n.species=n_distinct(spp.samples.DF$species)
      )
    }
  )
neutral.ASL <- llply(
  seq_along(unique(spp.samples.DF$species)),
  function(i) {
    make.single.species.AttributeSpace(
      site.data=select(grid.DF, contains(paste0(unique(spp.samples.DF$species)[i], '_neu
      species.data=select(filter(spp.samples.DF, species==unique(spp.samples.DF$species)
      spp.pos=i,
      n.species=n_distinct(spp.samples.DF$species)
    )
  }
)
# make table with targets
target.DF <- make.targets(
  species=unique(spp.samples.DF$species),
  environmental.space=as1[[1]], geographic.space=as1[[2]],
  adaptive.spaces=adaptive.ASL, neutral.spaces=neutral.ASL,
  amount.target=0.2, space.target=0.2
)
# make Rap objects
rd <- RapData(
  polygon=SpatialPolygons2PolySet(grid.PLY),
  pu=data.frame(
    cost=rep(1, nrow(grid.DF)),
    area=rep(1, nrow(grid.DF)),
    status=rep(0L, nrow(grid.DF))
  ),
  species=data.frame(name=unique(spp.samples.DF$species)),
  target=target.DF,
  attribute.spaces=append(append(surrogate.ASL, adaptive.ASL), neutral.ASL),
  pu.species.probabilities=ldply(
    seq_along(unique(spp.samples.DF$species)),
    .fun=function(i) {
      data.frame(
        species=i,
        pu=which(grid.DF[[unique(spp.samples.DF$species)[i]]]==1),
        value=1
      )
    }
  ),
  boundary=calcBoundaryData(grid.PLY)
)
ru <- RapUnsolved(RapUnreliableOpts(), rd)

```

```

## single species analysis
# generate RapSolved objects
single.spp.prioritisations <- llply(
  seq_along(unique(spp.samples.DF$species)),
  function(x) {
    llply(
      list(
        c(rapr.amount.target,0,0), c(rapr.amount.target,rapr.surrogate.target,0),
        c(rapr.amount.target,0,rapr.genetic.target,0)
      ),
      function(y) {
        species.prioritisation(
          x=spp.subset(ru, x),
          amount.targets=y[1],
          env.surrogate.targets=y[2],
          geo.surrogate.targets=y[2],
          adaptive.genetic.targets=y[3],
          neutral.genetic.targets=y[3],
          Threads=gb.Threads,
          MIPGap=gb.MIPGap
        )
      }
    )
  }
)

```

```

## Optimize a model with 1 rows, 388 columns and 45 nonzeros
## Coefficient statistics:
##   Matrix range      [1e+00, 1e+00]
##   Objective range   [1e+00, 1e+00]
##   Bounds range      [1e+00, 1e+00]
##   RHS range         [9e+00, 9e+00]
## Found heuristic solution: objective 9
## Presolve removed 1 rows and 388 columns
## Presolve time: 0.00s
## Presolve: All rows and columns removed
##
## Explored 0 nodes (0 simplex iterations) in 0.00 seconds
## Thread count was 1 (of 2 available processors)
##
## Optimal solution found (tolerance 9.00e-01)
## Best objective 9.000000000000e+00, best bound 9.000000000000e+00, gap 0.0%
## Optimize a model with 4143 rows, 4438 columns and 16245 nonzeros
## Coefficient statistics:
##   Matrix range      [1e-05, 8e+05]
##   Objective range   [1e+00, 1e+00]
##   Bounds range      [1e+00, 1e+00]

```



```

##   RHS range      [1e+00, 2e+06]
## Found heuristic solution: objective 40
## Presolve removed 0 rows and 343 columns
## Presolve time: 0.24s
## Presolved: 4143 rows, 4095 columns, 16245 nonzeros
## Variable types: 0 continuous, 4095 integer (4095 binary)
## Presolved: 4143 rows, 4095 columns, 16245 nonzeros
##
## Presolve removed 4051 rows and 200 columns
##
## Root relaxation: objective 9.000000e+00, 1401 iterations, 0.21 seconds
##
##      Nodes      |      Current Node      |      Objective Bounds      |      Work
##  Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time
##
## *    0      0              0      9.0000000    9.000000  0.00%    -    0s
##
## Explored 0 nodes (1448 simplex iterations) in 0.50 seconds
## Thread count was 1 (of 2 available processors)
##
## Optimal solution found (tolerance 9.00e-01)
## Best objective 9.000000000000e+00, best bound 9.000000000000e+00, gap 0.0%
## Optimize a model with 6028 rows, 6283 columns and 23625 nonzeros
## Coefficient statistics:
##   Matrix range      [1e-05, 1e+00]
##   Objective range [1e+00, 1e+00]
##   Bounds range      [1e+00, 1e+00]
##   RHS range         [1e+00, 1e+01]
## Presolve removed 0 rows and 343 columns
## Presolve time: 0.62s
## Presolved: 6028 rows, 5940 columns, 23496 nonzeros
## Variable types: 0 continuous, 5940 integer (5940 binary)
## Found heuristic solution: objective 45.0000000
## Found heuristic solution: objective 39.0000000
## Presolved: 6028 rows, 5940 columns, 23496 nonzeros
##
## Presolve removed 6025 rows and 5850 columns
##
## Root relaxation: objective 9.000000e+00, 1234 iterations, 0.23 seconds
##
##      Nodes      |      Current Node      |      Objective Bounds      |      Work
##  Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time
##
## *    0      0              0      9.0000000    9.000000  0.00%    -    0s
##
## Explored 0 nodes (1241 simplex iterations) in 0.95 seconds
## Thread count was 1 (of 2 available processors)
##

```

```

## Optimal solution found (tolerance 9.00e-01)
## Best objective 9.000000000000e+00, best bound 9.000000000000e+00, gap 0.0%
## Optimize a model with 1 rows, 388 columns and 129 nonzeros
## Coefficient statistics:
##   Matrix range      [1e+00, 1e+00]
##   Objective range   [1e+00, 1e+00]
##   Bounds range      [1e+00, 1e+00]
##   RHS range         [3e+01, 3e+01]
## Found heuristic solution: objective 26
## Presolve removed 1 rows and 388 columns
## Presolve time: 0.00s
## Presolve: All rows and columns removed
##
## Explored 0 nodes (0 simplex iterations) in 0.00 seconds
## Thread count was 1 (of 2 available processors)
##
## Optimal solution found (tolerance 9.00e-01)
## Best objective 2.600000000000e+01, best bound 2.600000000000e+01, gap 0.0%
## Optimize a model with 33543 rows, 33670 columns and 133257 nonzeros
## Coefficient statistics:
##   Matrix range      [1e-05, 9e+05]
##   Objective range   [1e+00, 1e+00]
##   Bounds range      [1e+00, 1e+00]
##   RHS range         [1e+00, 6e+06]
## Found heuristic solution: objective 114
## Presolve removed 0 rows and 259 columns (presolve time = 5s) ...
## Presolve removed 0 rows and 259 columns
## Presolve time: 6.19s
## Presolved: 33543 rows, 33411 columns, 133257 nonzeros
## Variable types: 0 continuous, 33411 integer (33411 binary)
## Presolved: 33543 rows, 33411 columns, 133257 nonzeros
##
## Presolve removed 33283 rows and 2012 columns
##
## Root simplex log...
##
## Iteration    Objective      Primal Inf.    Dual Inf.      Time
##          0      0.000000e+00    6.098853e+01    3.266558e+09     7s
##        3025    2.600000e+01    0.000000e+00    0.000000e+00     8s
##        3025    2.600000e+01    0.000000e+00    0.000000e+00     8s
##
## Root relaxation: objective 2.600000e+01, 3025 iterations, 2.17 seconds
##
##      Nodes   |   Current Node   |   Objective Bounds   |   Work
##  Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd  Gap | It/Node Time
##
##          0     0   26.00000    0    2  114.00000   26.00000   77.2%    -    8s
##

```

```

## Explored 0 nodes (5515 simplex iterations) in 8.85 seconds
## Thread count was 1 (of 2 available processors)
##
## Optimal solution found (tolerance 9.00e-01)
## Best objective 1.140000000000e+02, best bound 2.600000000000e+01, gap 77.1930%
## Optimize a model with 50052 rows, 50053 columns and 198789 nonzeros
## Coefficient statistics:
##   Matrix range      [1e-05, 1e+00]
##   Objective range   [1e+00, 1e+00]
##   Bounds range      [1e+00, 1e+00]
##   RHS range         [1e+00, 3e+01]
## Presolve removed 0 rows and 259 columns (presolve time = 5s) ...
## Presolve removed 0 rows and 259 columns (presolve time = 10s) ...
## Presolve removed 0 rows and 259 columns (presolve time = 15s) ...
## Presolve removed 0 rows and 259 columns
## Presolve time: 18.44s
## Presolved: 50052 rows, 49794 columns, 198436 nonzeros
## Variable types: 0 continuous, 49794 integer (49794 binary)
## Found heuristic solution: objective 113.0000000
## Found heuristic solution: objective 112.0000000
## Presolved: 50052 rows, 49794 columns, 198436 nonzeros
##
## Presolve removed 50019 rows and 45666 columns
##
## Root simplex log...
##
## Iteration    Objective      Primal Inf.    Dual Inf.      Time
##          0    1.2900000e+02  0.000000e+00  1.290000e+02   22s
##        2475    2.6000000e+01  0.000000e+00  0.000000e+00   23s
##        2475    2.6000000e+01  0.000000e+00  0.000000e+00   23s
##
## Root relaxation: objective 2.600000e+01, 2475 iterations, 4.01 seconds
##
##      Nodes   |   Current Node   |   Objective Bounds   |   Work
##  Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd  Gap | It/Node Time
##
## *    0      0              0    26.0000000  26.00000  0.00%   -   23s
##
## Explored 0 nodes (2778 simplex iterations) in 23.70 seconds
## Thread count was 1 (of 2 available processors)
##
## Optimal solution found (tolerance 9.00e-01)
## Best objective 2.600000000000e+01, best bound 2.600000000000e+01, gap 0.0%
## Optimize a model with 1 rows, 388 columns and 104 nonzeros
## Coefficient statistics:
##   Matrix range      [1e+00, 1e+00]
##   Objective range   [1e+00, 1e+00]
##   Bounds range      [1e+00, 1e+00]

```

```

##   RHS range      [2e+01, 2e+01]
## Found heuristic solution: objective 21
## Presolve removed 1 rows and 388 columns
## Presolve time: 0.00s
## Presolve: All rows and columns removed
##
## Explored 0 nodes (0 simplex iterations) in 0.00 seconds
## Thread count was 1 (of 2 available processors)
##
## Optimal solution found (tolerance 9.00e-01)
## Best objective 2.100000000000e+01, best bound 2.100000000000e+01, gap 0.0%
## Optimize a model with 21843 rows, 22020 columns and 86632 nonzeros
## Coefficient statistics:
##   Matrix range      [1e-05, 8e+05]
##   Objective range   [1e+00, 1e+00]
##   Bounds range      [1e+00, 1e+00]
##   RHS range         [1e+00, 4e+06]
## Found heuristic solution: objective 98
## Presolve removed 0 rows and 284 columns
## Presolve time: 3.11s
## Presolved: 21843 rows, 21736 columns, 86632 nonzeros
## Variable types: 0 continuous, 21736 integer (21736 binary)
## Presolved: 21843 rows, 21736 columns, 86632 nonzeros
##
## Presolve removed 21633 rows and 1637 columns
##
## Root relaxation: objective 2.100000e+01, 2175 iterations, 1.08 seconds
##
##   Nodes   |   Current Node   |   Objective Bounds   |   Work
##  Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd   Gap | It/Node Time
##
## *    0     0               0      21.0000000    21.000000  0.00%    -    4s
##
## Explored 0 nodes (2182 simplex iterations) in 4.29 seconds
## Thread count was 1 (of 2 available processors)
##
## Optimal solution found (tolerance 9.00e-01)
## Best objective 2.100000000000e+01, best bound 2.100000000000e+01, gap 0.0%
## Optimize a model with 32237 rows, 32316 columns and 127816 nonzeros
## Coefficient statistics:
##   Matrix range      [1e-03, 1e+00]
##   Objective range   [1e+00, 1e+00]
##   Bounds range      [1e+00, 1e+00]
##   RHS range         [1e+00, 2e+01]
## Presolve removed 0 rows and 284 columns (presolve time = 5s) ...
## Presolve removed 0 rows and 284 columns (presolve time = 10s) ...
## Presolve removed 0 rows and 284 columns
## Presolve time: 10.63s

```

```

## Presolved: 32237 rows, 32032 columns, 127509 nonzeros
## Variable types: 0 continuous, 32032 integer (32032 binary)
## Found heuristic solution: objective 94.0000000
## Presolved: 32237 rows, 32032 columns, 127509 nonzeros
##
## Presolve removed 32237 rows and 32032 columns
##
## Root simplex log...
##
## Iteration      Objective      Primal Inf.      Dual Inf.      Time
##      0      1.0400000e+02      0.000000e+00      1.040000e+02      12s
##     1929      2.1000000e+01      0.000000e+00      0.000000e+00      13s
##     1929      2.1000000e+01      0.000000e+00      0.000000e+00      13s
##
## Root relaxation: objective 2.100000e+01, 1929 iterations, 1.85 seconds
##
##      Nodes      |      Current Node      |      Objective Bounds      |      Work
##  Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd    Gap | It/Node Time
##
## *    0      0              0      21.0000000    21.000000    0.00%      -   14s
##
## Explored 0 nodes (3879 simplex iterations) in 14.14 seconds
## Thread count was 1 (of 2 available processors)
##
## Optimal solution found (tolerance 9.00e-01)
## Best objective 2.1000000000000e+01, best bound 2.1000000000000e+01, gap 0.0%

```

```
# generate results table
```

```

single.spp.DF <- ldply(
  single.spp.prioritisations,
  function(x) {
    mutate(
      ldply(x, extractResults),
      Prioritisation=c('Amount', 'Surrogate', 'Genetic')
    )
  }
)

```

```
## multispecies analysis
```

```
# make prioritisations
```

```

multi.spp.prioritisations <- llply(
  list(
    c(rapr.amount.target,0,0), c(rapr.amount.target,rapr.surrogate.target,0),
    c(rapr.amount.target,0,rapr.genetic.target,0)
  ),
  function(y) {
    species.prioritisation(

```

```

        x=ru,
        amount.targets=y[1],
        env.surrogate.targets=y[2],
        geo.surrogate.targets=y[2],
        adaptive.genetic.targets=y[3],
        neutral.genetic.targets=y[3],
        Threads=gb.Threads,
        MIPGap=gb.MIPGap
    )
}
)

```

```

## Optimize a model with 3 rows, 388 columns and 278 nonzeros
## Coefficient statistics:
##   Matrix range      [1e+00, 1e+00]
##   Objective range   [1e+00, 1e+00]
##   Bounds range      [1e+00, 1e+00]
##   RHS range         [9e+00, 3e+01]
## Found heuristic solution: objective 29
## Presolve removed 3 rows and 388 columns
## Presolve time: 0.00s
## Presolve: All rows and columns removed
##
## Explored 0 nodes (0 simplex iterations) in 0.00 seconds
## Thread count was 1 (of 2 available processors)
##
## Optimal solution found (tolerance 9.00e-01)
## Best objective 2.6000000000000e+01, best bound 2.6000000000000e+01, gap 0.0%
## Optimize a model with 59529 rows, 59352 columns and 236134 nonzeros
## Coefficient statistics:
##   Matrix range      [1e-05, 9e+05]
##   Objective range   [1e+00, 1e+00]
##   Bounds range      [1e+00, 1e+00]
##   RHS range         [1e+00, 6e+06]
## Found heuristic solution: objective 141
## Presolve removed 0 rows and 242 columns (presolve time = 5s) ...
## Presolve removed 0 rows and 242 columns (presolve time = 10s) ...
## Presolve removed 0 rows and 242 columns (presolve time = 15s) ...
## Presolve removed 0 rows and 242 columns
## Presolve time: 17.25s
## Presolved: 59529 rows, 59110 columns, 236134 nonzeros
## Variable types: 0 continuous, 59110 integer (59110 binary)
## Presolved: 59529 rows, 59110 columns, 236134 nonzeros
##
## Presolve removed 58967 rows and 3717 columns
##
## Root simplex log...

```

```

##
## Iteration      Objective      Primal Inf.      Dual Inf.      Time
##      0      0.0000000e+00      1.432025e+02      5.292865e+09      20s
##      1911      8.6975993e+01      0.000000e+00      4.185188e+03      20s
##      8122      2.9997714e+01      0.000000e+00      1.732555e+04      25s
##      11032      2.6000000e+01      0.000000e+00      0.000000e+00      29s
##      11032      2.6000000e+01      0.000000e+00      0.000000e+00      29s
##
## Root relaxation: objective 2.600000e+01, 11032 iterations, 12.06 seconds
## Total elapsed time = 30.45s
##
##      Nodes      |      Current Node      |      Objective Bounds      |      Work
## Expl Unexpl | Obj Depth IntInf | Incumbent      BestBd      Gap | It/Node Time
##
##      0      0      26.00000      0      228      141.00000      26.00000      81.6%      -      33s
##
## Explored 0 nodes (16236 simplex iterations) in 33.64 seconds
## Thread count was 1 (of 2 available processors)
##
## Optimal solution found (tolerance 9.00e-01)
## Best objective 1.4100000000000e+02, best bound 2.6000000000000e+01, gap 81.5603%
## Optimize a model with 88317 rows, 87876 columns and 350230 nonzeros
## Coefficient statistics:
##   Matrix range      [1e-05, 1e+00]
##   Objective range [1e+00, 1e+00]
##   Bounds range      [1e+00, 1e+00]
##   RHS range         [1e+00, 3e+01]
## Presolve removed 0 rows and 242 columns (presolve time = 5s) ...
## Presolve removed 0 rows and 242 columns (presolve time = 10s) ...
## Presolve removed 0 rows and 242 columns (presolve time = 15s) ...
## Presolve removed 0 rows and 242 columns (presolve time = 20s) ...
## Presolve removed 0 rows and 242 columns (presolve time = 25s) ...
## Presolve removed 0 rows and 242 columns (presolve time = 30s) ...
## Presolve removed 0 rows and 242 columns (presolve time = 35s) ...
## Presolve removed 0 rows and 242 columns (presolve time = 40s) ...
## Presolve removed 0 rows and 242 columns (presolve time = 45s) ...
## Presolve removed 0 rows and 242 columns (presolve time = 50s) ...
## Presolve removed 0 rows and 242 columns (presolve time = 55s) ...
## Presolve removed 0 rows and 242 columns
## Presolve time: 58.62s
## Presolved: 88317 rows, 87634 columns, 349441 nonzeros
## Variable types: 0 continuous, 87634 integer (87634 binary)
## Found heuristic solution: objective 139.0000000
## Presolved: 88317 rows, 87634 columns, 349441 nonzeros
##
## Presolve removed 88281 rows and 83416 columns
##
## Root simplex log...

```

```
##
## Iteration      Objective      Primal Inf.      Dual Inf.      Time
##           0      1.4600000e+02  0.000000e+00    1.460000e+02    63s
##          2069      6.2993159e+01  0.000000e+00    5.682674e+03    65s
##          5280      3.7092713e+01  0.000000e+00    1.835344e+04    70s
##          8491      2.6605111e+01  0.000000e+00    3.177995e+03    75s
##          9559      2.6000000e+01  0.000000e+00    0.000000e+00    80s
##          9559      2.6000000e+01  0.000000e+00    0.000000e+00    80s
##
## Root relaxation: objective 2.600000e+01, 9559 iterations, 19.52 seconds
## Total elapsed time = 80.01s
##
##      Nodes      |      Current Node      |      Objective Bounds      |      Work
##  Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd    Gap | It/Node Time
##
## *    0      0              0      26.0000000    26.000000  0.00%    -    80s
##
## Explored 0 nodes (13085 simplex iterations) in 80.67 seconds
## Thread count was 1 (of 2 available processors)
##
## Optimal solution found (tolerance 9.00e-01)
## Best objective 2.600000000000e+01, best bound 2.600000000000e+01, gap 0.0%
```

```
# generate results table
multi.spp.DF <- ldply(seq_along(multi.spp.prioritisations), function(i) {
  mutate(
    extractResults(multi.spp.prioritisations[[i]]),
    Prioritisation=c('Amount', 'Surrogate', 'Genetic')[i]
  )
})
```

```
## pareto frontier analysis
# generate prioritistions
env.pareto.prioritisations <- llply(
  rapr.pareto.surrogate.targets,
  species.prioritisation,
  x=ru,
  amount.targets=0,
  geo.surrogate.targets=0,
  adaptive.genetic.targets=0,
  neutral.genetic.targets=0
)
```

```
## Optimize a model with 29766 rows, 29870 columns and 118206 nonzeros
## Coefficient statistics:
##   Matrix range      [1e-05, 2e+03]
##   Objective range   [1e+00, 1e+00]
```



```

## Bounds range [1e+00, 1e+00]
## RHS range [1e+00, 2e+05]
## Found heuristic solution: objective 118
## Presolve removed 3 rows and 242 columns
## Presolve time: 4.71s
## Presolved: 29763 rows, 29628 columns, 117928 nonzeros
## Variable types: 0 continuous, 29628 integer (29628 binary)
## Presolved: 29763 rows, 29628 columns, 117928 nonzeros
##
## Presolve removed 29482 rows and 146 columns
##
## Root simplex log...
##
## Iteration Objective Primal Inf. Dual Inf. Time
## 0 0.0000000e+00 2.530709e-01 9.212604e+07 6s
## 5544 1.0000000e+00 0.000000e+00 0.000000e+00 10s
## 5544 1.0000000e+00 0.000000e+00 0.000000e+00 10s
##
## Root relaxation: objective 1.000000e+00, 5544 iterations, 5.04 seconds
##
## Nodes | Current Node | Objective Bounds | Work
## Expl Unexpl | Obj Depth IntInf | Incumbent BestBd Gap | It/Node Time
##
## * 0 0 0 1.0000000 1.000000 0.00% - 9s
##
## Explored 0 nodes (5544 simplex iterations) in 9.89 seconds
## Thread count was 1 (of 2 available processors)
##
## Optimal solution found (tolerance 5.00e-02)
## Best objective 1.000000000000e+00, best bound 1.000000000000e+00, gap 0.0%
## Optimize a model with 29766 rows, 29870 columns and 118206 nonzeros
## Coefficient statistics:
## Matrix range [1e-05, 2e+03]
## Objective range [1e+00, 1e+00]
## Bounds range [1e+00, 1e+00]
## RHS range [1e+00, 8e+04]
## Found heuristic solution: objective 118
## Presolve removed 3 rows and 242 columns
## Presolve time: 4.76s
## Presolved: 29763 rows, 29628 columns, 117928 nonzeros
## Variable types: 0 continuous, 29628 integer (29628 binary)
## Presolved: 29763 rows, 29628 columns, 117928 nonzeros
##
## Presolve removed 29482 rows and 146 columns
##
## Root simplex log...
##
## Iteration Objective Primal Inf. Dual Inf. Time

```

```

##          0      0.0000000e+00      2.368537e+01      1.976974e+09      6s
##      16076      1.9945643e+00      0.000000e+00      2.476667e+02      10s
##      24782      1.0000000e+00      0.000000e+00      0.000000e+00      14s
##      24782      1.0000000e+00      0.000000e+00      0.000000e+00      14s
##
## Root relaxation: objective 1.000000e+00, 24782 iterations, 9.57 seconds
##
##      Nodes      |      Current Node      |      Objective Bounds      |      Work
##  Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd    Gap | It/Node Time
##
##          0      0      1.00000      0  558  118.00000      1.00000  99.2%    -   14s
## H          0      0                          1.0000000      1.00000  0.00%    -   15s
##
## Explored 0 nodes (28177 simplex iterations) in 15.19 seconds
## Thread count was 1 (of 2 available processors)
##
## Optimal solution found (tolerance 5.00e-02)
## Best objective 1.000000000000e+00, best bound 1.000000000000e+00, gap 0.0%
## Optimize a model with 29766 rows, 29870 columns and 118206 nonzeros
## Coefficient statistics:
##   Matrix range      [1e-05, 2e+03]
##   Objective range   [1e+00, 1e+00]
##   Bounds range      [1e+00, 1e+00]
##   RHS range         [5e-04, 1e+00]
## Found heuristic solution: objective 146
## Presolve removed 29766 rows and 29870 columns
## Presolve time: 0.06s
## Presolve: All rows and columns removed
##
## Explored 0 nodes (0 simplex iterations) in 0.08 seconds
## Thread count was 1 (of 2 available processors)
##
## Optimal solution found (tolerance 5.00e-02)
## Best objective 1.460000000000e+02, best bound 1.460000000000e+02, gap 0.0%

```

```

geo.pareto.prioritisations <- llply(
  rapr.pareto.surrogate.targets,
  species.prioritisation,
  x=ru,
  amount.targets=0,
  env.surrogate.targets=0,
  adaptive.genetic.targets=0,
  neutral.genetic.targets=0
)

```

```

## Optimize a model with 29766 rows, 29870 columns and 118206 nonzeros
## Coefficient statistics:

```

```

## Matrix range [1e-05, 9e+05]
## Objective range [1e+00, 1e+00]
## Bounds range [1e+00, 1e+00]
## RHS range [1e+00, 6e+07]
## Found heuristic solution: objective 118
## Presolve removed 3 rows and 242 columns
## Presolve time: 4.67s
## Presolved: 29763 rows, 29628 columns, 117928 nonzeros
## Variable types: 0 continuous, 29628 integer (29628 binary)
## Presolved: 29763 rows, 29628 columns, 117928 nonzeros
##
## Presolve removed 29482 rows and 3717 columns
##
## Root simplex log...
##
## Iteration Objective Primal Inf. Dual Inf. Time
## 0 0.0000000e+00 1.897425e-01 2.726995e+08 6s
## 5754 1.0000000e+00 0.000000e+00 0.000000e+00 10s
## 5754 1.0000000e+00 0.000000e+00 0.000000e+00 10s
##
## Root relaxation: objective 1.000000e+00, 5754 iterations, 5.17 seconds
##
## Nodes | Current Node | Objective Bounds | Work
## Expl Unexpl | Obj Depth IntInf | Incumbent BestBd Gap | It/Node Time
##
## * 0 0 0 1.0000000 1.00000 0.00% - 9s
##
## Explored 0 nodes (5754 simplex iterations) in 9.95 seconds
## Thread count was 1 (of 2 available processors)
##
## Optimal solution found (tolerance 5.00e-02)
## Best objective 1.000000000000e+00, best bound 1.000000000000e+00, gap 0.0%
## Optimize a model with 29766 rows, 29870 columns and 118206 nonzeros
## Coefficient statistics:
## Matrix range [1e-05, 9e+05]
## Objective range [1e+00, 1e+00]
## Bounds range [1e+00, 1e+00]
## RHS range [1e+00, 3e+07]
## Found heuristic solution: objective 124
## Presolve removed 3 rows and 242 columns
## Presolve time: 4.77s
## Presolved: 29763 rows, 29628 columns, 117928 nonzeros
## Variable types: 0 continuous, 29628 integer (29628 binary)
## Presolved: 29763 rows, 29628 columns, 117928 nonzeros
##
## Presolve removed 29482 rows and 3717 columns
##
## Root simplex log...

```

```

##
## Iteration      Objective      Primal Inf.      Dual Inf.      Time
##           0      0.000000e+00      3.367941e+01      3.315739e+09      6s
##       11511      3.251977e+00      0.000000e+00      1.688364e+03      10s
##       15567      3.065492e+00      0.000000e+00      5.864860e+02      15s
##       19103      2.911759e+00      0.000000e+00      3.285617e+02      20s
##       22327      2.786547e+00      0.000000e+00      3.669895e+02      25s
##       25447      2.643107e+00      0.000000e+00      5.631065e+02      30s
##       28671      2.476370e+00      0.000000e+00      7.578635e+02      35s
##       30335      2.441423e+00      0.000000e+00      1.019895e+03      40s
##       31479      2.388428e+00      0.000000e+00      1.015067e+03      45s
##       32623      2.303487e+00      0.000000e+00      8.756853e+02      50s
##       33767      2.237032e+00      0.000000e+00      1.446064e+03      55s
##       35015      2.155209e+00      0.000000e+00      3.573627e+02      60s
##       36159      2.088120e+00      0.000000e+00      4.980702e+02      65s
##       37303      2.029669e+00      0.000000e+00      1.626541e+03      70s
##       38447      1.983838e+00      0.000000e+00      5.277967e+02      75s
##       39487      1.946619e+00      0.000000e+00      2.551990e+02      80s
##       40631      1.905230e+00      0.000000e+00      6.215684e+03      85s
##       41671      1.864488e+00      0.000000e+00      3.198452e+02      90s
##       42815      1.806298e+00      0.000000e+00      6.754032e+02      95s
##       43751      1.754482e+00      0.000000e+00      7.438513e+03      100s
##       44791      1.707428e+00      0.000000e+00      1.461999e+03      105s
##       45727      1.666089e+00      0.000000e+00      5.164360e+03      110s
##       46767      1.622197e+00      0.000000e+00      2.018706e+03      115s
##       47703      1.596980e+00      0.000000e+00      1.349636e+03      120s
##       48743      1.577046e+00      0.000000e+00      3.226639e+02      125s
##       49783      1.565499e+00      0.000000e+00      2.599145e+02      130s
##       50927      1.553614e+00      0.000000e+00      9.848115e+02      135s
##       51863      1.545910e+00      0.000000e+00      1.069567e+03      140s
##       52903      1.541007e+00      0.000000e+00      4.231634e+02      145s
##       53839      1.535524e+00      0.000000e+00      2.467873e+02      150s
##       54775      1.529304e+00      0.000000e+00      5.075302e+02      155s
##       55711      1.517959e+00      0.000000e+00      1.158771e+03      160s
##       56647      1.498111e+00      0.000000e+00      6.710989e+02      165s
##       57583      1.478374e+00      0.000000e+00      3.841066e+02      170s
##       58519      1.462060e+00      0.000000e+00      5.926053e+02      175s
##       59455      1.439496e+00      0.000000e+00      3.074928e+03      180s
##       60287      1.421758e+00      0.000000e+00      3.953825e+03      185s
##       61223      1.395823e+00      0.000000e+00      1.524913e+03      190s
##       62159      1.361299e+00      0.000000e+00      2.674519e+03      195s
##       63095      1.286862e+00      0.000000e+00      1.717281e+04      200s
##       64759      1.014303e+00      0.000000e+00      7.681355e+03      205s
##       66735      9.956667e-01      0.000000e+00      6.845464e+02      210s
##       67107      1.000000e+00      0.000000e+00      0.000000e+00      211s
##       67107      1.000000e+00      0.000000e+00      0.000000e+00      211s
##
## Root relaxation: objective 1.000000e+00, 67107 iterations, 205.90 seconds

```

```
##
##      Nodes      |      Current Node      |      Objective Bounds      |      Work
##  Expl Unexpl |  Obj  Depth IntInf | Incumbent    BestBd    Gap | It/Node Time
##
## *      0      0              0      1.0000000    1.000000  0.00%    -   210s
##
## Explored 0 nodes (67107 simplex iterations) in 210.79 seconds
## Thread count was 1 (of 2 available processors)
##
## Optimal solution found (tolerance 5.00e-02)
## Best objective 1.000000000000e+00, best bound 1.000000000000e+00, gap 0.0%
## Optimize a model with 29766 rows, 29870 columns and 118206 nonzeros
## Coefficient statistics:
##   Matrix range      [1e-05, 9e+05]
##   Objective range   [1e+00, 1e+00]
##   Bounds range      [1e+00, 1e+00]
##   RHS range         [5e-04, 1e+00]
## Found heuristic solution: objective 146
## Presolve removed 29766 rows and 29870 columns
## Presolve time: 0.06s
## Presolve: All rows and columns removed
##
## Explored 0 nodes (0 simplex iterations) in 0.08 seconds
## Thread count was 1 (of 2 available processors)
##
## Optimal solution found (tolerance 5.00e-02)
## Best objective 1.460000000000e+02, best bound 1.460000000000e+02, gap 0.0%
```

```
# extract results
env.pareto.DF <- ldply(seq_along(env.pareto.prioritisations), function(i) {
  mutate(
    extractResults(env.pareto.prioritisations[[i]]),
    Surrogate.target=rapr.pareto.surrogate.targets[i]
  )
})
geo.pareto.DF <- ldply(seq_along(geo.pareto.prioritisations), function(i) {
  mutate(
    extractResults(geo.pareto.prioritisations[[i]]),
    Surrogate.target=rapr.pareto.surrogate.targets[i]
  )
})
```

## Results

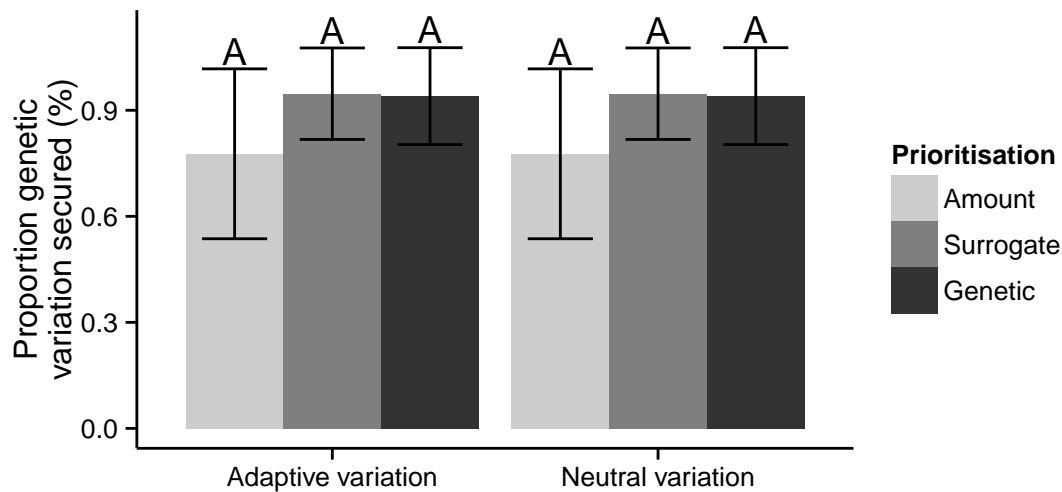
### Single species prioritisations

```
## statistical analysis
# prepare data
single.spp.SDF <- single.spp.DF %>%
  gather(Metric, value, amount.held:neutral.held) %>%
  filter(Metric %in% c('adaptive.held', 'neutral.held')) %>%
  mutate(Metric=revalue(Metric, c('adaptive.held'='Adaptive variation',
    'neutral.held'='Neutral variation')))) %>%
  mutate(Prioritisation.Metric=interaction(Prioritisation,Metric))
# model
single.spp.GLM <- suppressWarnings(glm(value ~ Prioritisation * Metric,
  family='binomial', data=single.spp.SDF))
single.spp.AOV <- suppressWarnings(anova(single.spp.GLM))
# post-hoc
single.spp.GLM2 <- suppressWarnings(glm(value ~ Prioritisation.Metric,
  family='binomial', data=single.spp.SDF))
single.spp.MCP <- summary(
  glht(single.spp.GLM2,
    linct=mcp(Prioritisation.Metric='Tukey')),
  adjusted('bonferroni'))

# prepare data for plotting
single.spp.PDF <- expand.grid(
  Prioritisation=unique(single.spp.SDF$Prioritisation),
  Metric=unique(single.spp.SDF$Metric))
single.spp.PDF <- cbind(single.spp.PDF,
  as.data.frame(predict(single.spp.GLM, single.spp.PDF,
    type='response', se.fit=TRUE))) %>%
  mutate(lower=fit-se.fit, upper=fit+se.fit,
    letters=toupper(cld(single.spp.MCP)$mcletters$Letters),
    letter_pos=upper+0.05)
# make plot
ggplot(aes(x=Metric,y=fit,fill=Prioritisation),
  data=single.spp.PDF) +
  geom_bar(position=position_dodge(0.9),
    stat='identity') +
  geom_errorbar(
    aes(ymin=lower,ymax=upper),
    position=position_dodge(0.9), width=0.6) +
  geom_text(aes(x=Metric, y=letter_pos,
    label=letters), position=position_dodge(0.9)) +
  scale_fill_manual(name='Prioritisation',
    values=c('grey80','grey50','grey20')) +
```

```
ylab('Proportion genetic\nvariation secured (%)') +
xlab('') +
theme_classic()
```

```
## ymax not defined: adjusting position using y instead
```



**Figure 1** Summary of single species prioritisations. Single species prioritisations were generated using amount-based targets, amount-based and surrogate-based targets, and amount-based and genetic-based targets for each species. Data shows the performance of prioritisations generated using these three sets of targets. Bars denote means and standard errors.

## Multi-species prioritisations

```
## statistical analysis
# prepare data
multi.spp.SDF <- multi.spp.DF %>%
  gather(Metric, value, amount.held:neutral.held) %>%
  filter(Metric %in% c('adaptive.held', 'neutral.held')) %>%
  mutate(Metric=revalue(Metric, c('adaptive.held'='Adaptive variation',
    'neutral.held'='Neutral variation')))) %>%
  mutate(Prioritisation.Metric=interaction(Prioritisation,Metric))
# model
multi.spp.GLM <- suppressWarnings(glm(value ~ Prioritisation * Metric,
  family='binomial', data=multi.spp.SDF))
multi.spp.AOV <- suppressWarnings(anova(multi.spp.GLM))
# post-hoc
multi.spp.GLM2 <- suppressWarnings(glm(value ~ Prioritisation.Metric,
  family='binomial', data=multi.spp.SDF))
```

```

multi.spp.MCP <- summary(
  glht(multi.spp.GLM2,
    linfct=mcp(Prioritisation.Metric='Tukey')),
  adjusted('bonferroni'))

# download basemap
data(countriesHigh)
countries.FPLY <- countriesHigh[
  countriesHigh$ADMIN %in% c(
    'Italy', 'Switzerland', 'France', 'Austria',
    'Germany', 'Slovenia', 'Croatia', 'Hungary',
    'Monaco', 'Germany'
  )
,] %>% spFortify
# prepare data for plotting
multi.spp.grid.FPLY <- grid.PLY
for (i in seq_along(multi.spp.prioritisations))
  multi.spp.grid.FPLY@data[[paste0('v',i)]] <- selections(multi.spp.prioritisations[[i]])
multi.spp.grid.FPLY <- spFortify(multi.spp.grid.FPLY)
# make maps
do.call(
  grid.arrange,
  append(
    llply(
      seq_along(multi.spp.prioritisations),
      function(i) {
        ggplot() +
          geom_polygon(data=countries.FPLY, aes(x=long, y=lat, group=group),
            fill='grey20', color='grey80') +
          geom_polygon(data=multi.spp.grid.FPLY, aes_string(x='long', y='lat',
            group='group', fill=paste0('v',i)),
            alpha=0.8, color='grey10') +
          guides(fill=guide_legend(title=' ')) +
          theme_classic() +
          theme(axis.ticks=element_blank(), axis.text=element_blank(),
            plot.margin=unit(c(0,0,0,0),'cm'), axis.line=element_blank(),
            legend.position='none') +
          coord_cartesian(
            xlim=buffered.range(multi.spp.grid.FPLY$long, 0.05),
            ylim=buffered.range(multi.spp.grid.FPLY$lat, 0.05)
          ) +
          xlab('') +
          ylab('') +
          annotate('text',
            x=min(multi.spp.grid.FPLY$long)+diff(range(multi.spp.grid.FPLY$long))*0.05,
            y=min(multi.spp.grid.FPLY$lat)+diff(range(multi.spp.grid.FPLY$lat))*1.05,
            label=letters[i], hjust=1, vjust=1, color='white', size=8)
      }
    )
  )
)

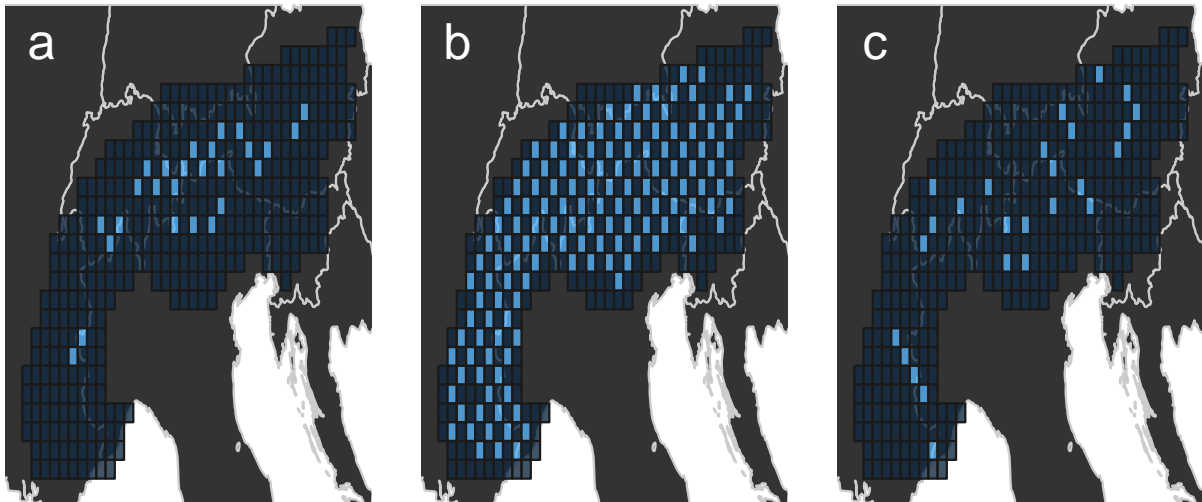
```



```

    }
  ),
  list(nrow=1)
)
)

```



**Figure 2** Multi-species prioritisations. Panel (a) shows the prioritisation generated for using just amount-based targets. Panel (b) shows the prioritisation generated using amount-based and surrogate based targets. Panel (c) shows the prioritisation generated using amount-based and genetic-based targets

```

# prepare data for plotting
multi.spp.PDF <- expand.grid(
  Prioritisation=unique(multi.spp.SDF$Prioritisation),
  Metric=unique(multi.spp.SDF$Metric))
multi.spp.PDF <- cbind(multi.spp.PDF,
  as.data.frame(predict(multi.spp.GLM, multi.spp.PDF,
    type='response', se.fit=TRUE))) %>%
  mutate(lower=fit-se.fit, upper=fit+se.fit,
    letters=toupper(cld(multi.spp.MCP)$mcletters$Letters),
    letter_pos=upper+0.05)
# make plot
ggplot(aes(x=Metric,y=fit,fill=Prioritisation),
  data=multi.spp.PDF) +
  geom_bar(position=position_dodge(0.9),
    stat='identity') +
  geom_errorbar(
    aes(ymin=lower,ymax=upper),
    position=position_dodge(0.9), width=0.6) +
  geom_text(aes(x=Metric, y=letter_pos,
    label=letters), position=position_dodge(0.9)) +
  scale_fill_manual(name='Prioritisation',

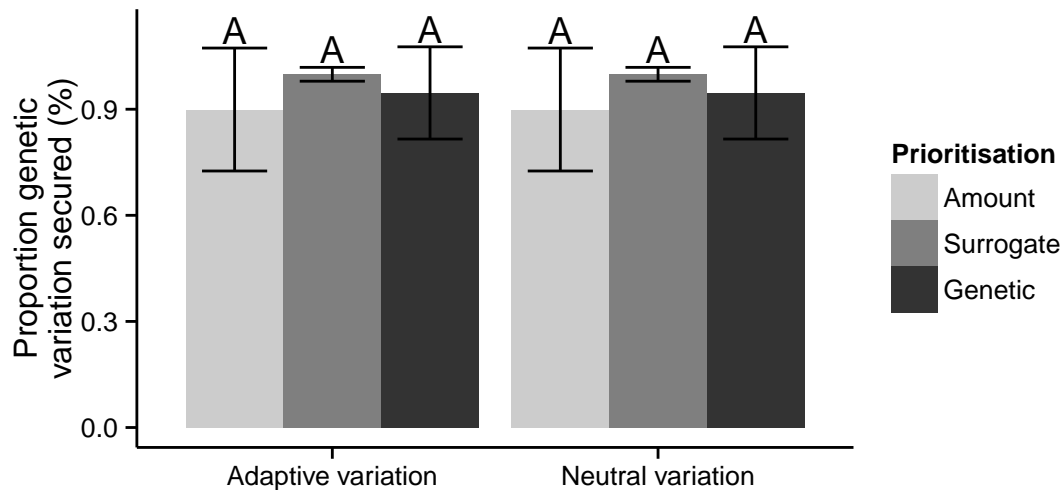
```

```

    values=c('grey80','grey50','grey20')) +
  ylab('Proportion genetic\nvariation secured (%)') +
  xlab('') +
  theme_classic()

```

## ymax not defined: adjusting position using y instead



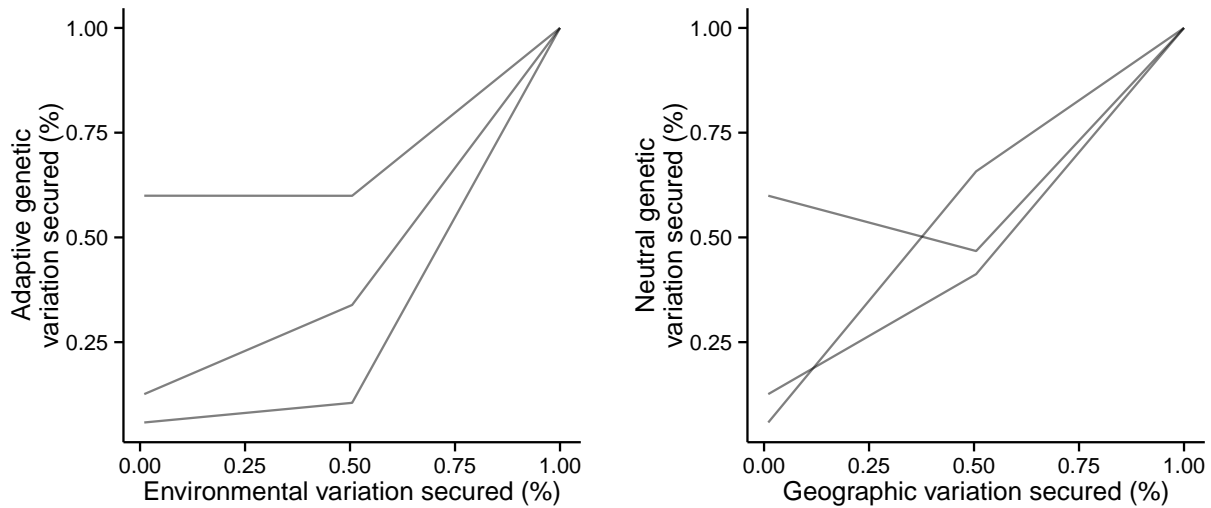
**Figure 3** Summary of multi-species prioritisations. Three prioritisations were generated using amount-based targets, amount-based and surrogate-based targets, and amount-based and genetic-based targets for each species. Data shows the performance of these prioritisations based on how much genetic variation they explain. Bars denote means and standard errors.

## Pareto-frontier analysis

```

# make plots
p1 <- ggplot(data=env.pareto.DF) +
  geom_line(aes(x=Surrogate.target,y=adaptive.held,group=Species),
    alpha=0.5) +
  xlab('Environmental variation secured (%)') +
  ylab('Adaptive genetic\nvariation secured (%)') +
  theme_classic()
p2 <- ggplot(data=geo.pareto.DF) +
  geom_line(aes(x=Surrogate.target,y=neutral.held,group=Species),
    alpha=0.5) +
  xlab('Geographic variation secured (%)') +
  ylab('Neutral genetic\nvariation secured (%)') +
  theme_classic()
grid.arrange(p1, p2, nrow=1)

```



**Figure 4** The relationship between surrogates and genetic variation secured in prioritisations.

## Discussion

## Acknowledgements

JOH is funded by an Australian Postgraduate Award (APA) scholarship. RAF has an Australian Research Council Future Fellowship. This work was supported by the Centre of Excellence for Environmental Decisions (CEED) and the Landscape Ecology and Conservation Group (LEC) at The University of Queensland.

## References

## Supporting Information

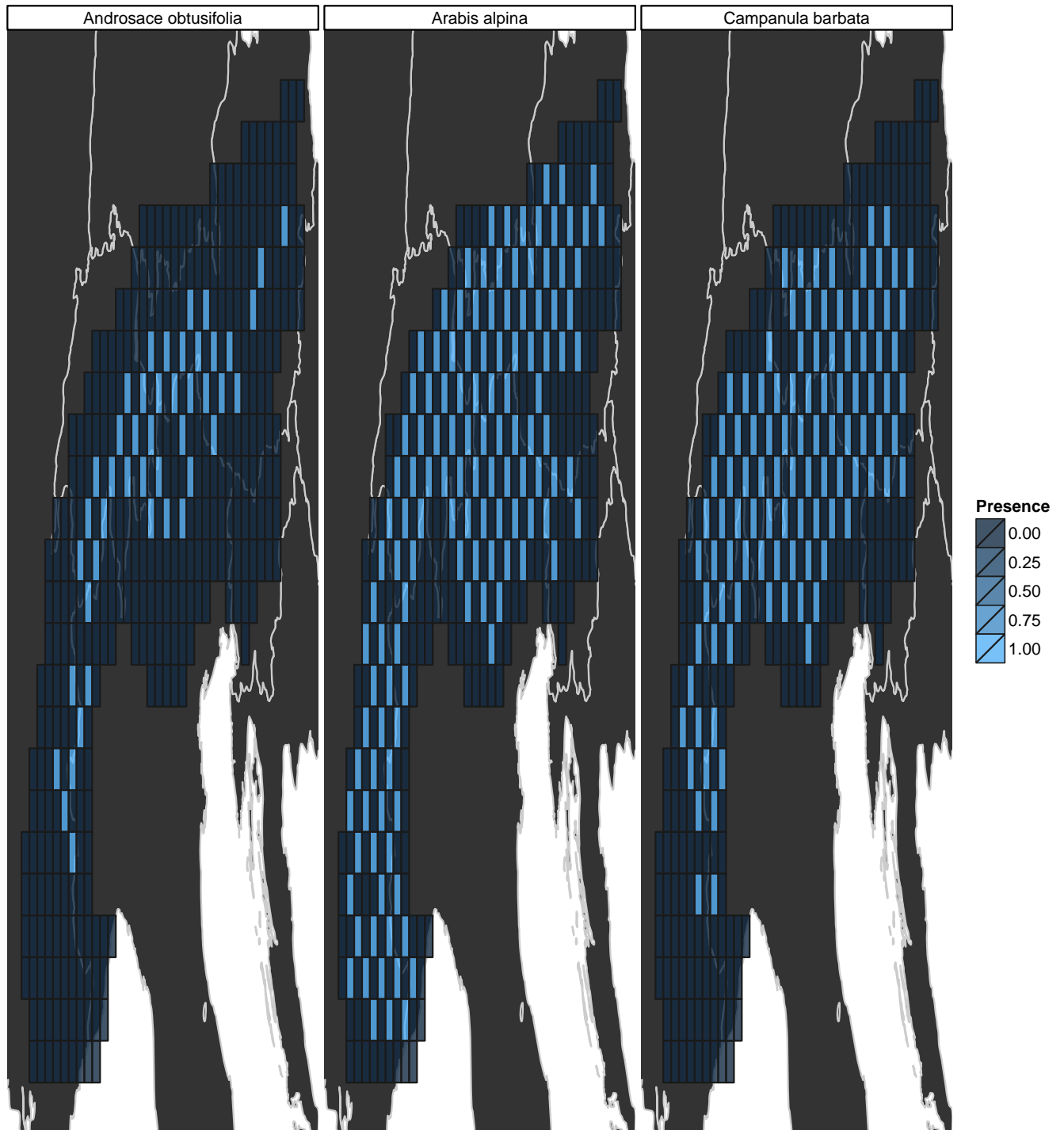
### Appendix S1: Species distributions

```
## plot map of species distributions
# fortify data
grid.FPLY <- spFortify(grid.PLY)
spp.grid.FPLY <- ldply(unique(spp.samples.DF$species), function(x) {
  z <- grid.FPLY[,c('long', 'lat', 'group', x),drop=FALSE]
  names(z)[4] <- 'presence'
  z$species <- gsub('\\_ ', ' ', x)
  return(z)
})
# plot species data
```

```

ggplot() +
  geom_polygon(data=countries.FPLY, aes(x=long, y=lat, group=group),
    fill='grey20', color='grey80') +
  geom_polygon(data=spp.grid.FPLY, aes(x=long, y=lat,
    group=group, fill=presence), alpha=0.8, color='grey10') +
  theme_classic() +
  guides(fill=guide_legend(title='Presence')) +
  theme(axis.ticks=element_blank(), axis.text=element_blank(),
    axis.line=element_blank()) +
  coord_cartesian(
    xlim=buffered.range(grid.FPLY$long, 0.05),
    ylim=buffered.range(grid.FPLY$lat, 0.05)
  ) +
  xlab('') +
  ylab('') +
  facet_wrap(~ species, ncol=4)

```



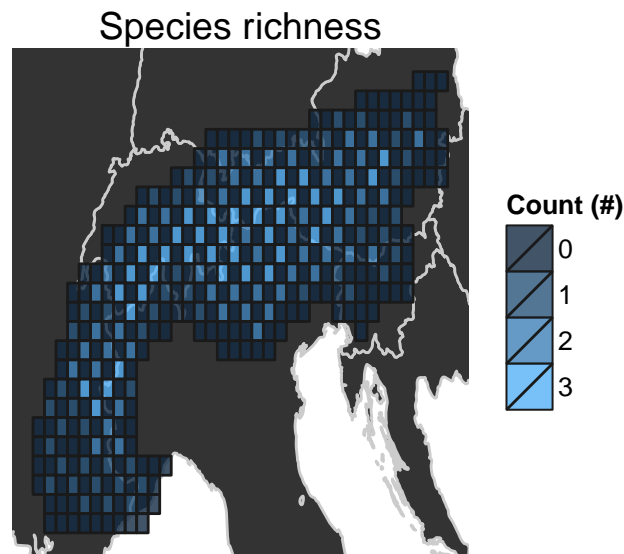
**Figure 5** Species distributions. Squares represent planning units. For a given species, planning units that were found to be inhabited are denoted with bright blue.

```

# calculate species richness
grid.PLY$Species_richness <- grid.PLY@data %>%
  select(5:(4+n.spp)) %>% as.matrix() %>% rowSums()

# plot species richness
ggplot() +
  geom_polygon(data=countries.FPLY, aes(x=long, y=lat, group=group),
    fill='grey20', color='grey80') +
  geom_polygon(data=spFortify(grid.PLY), aes(x=long, y=lat,
    group=group, fill=Species_richness), alpha=0.8, color='grey10') +
  guides(fill=guide_legend(title='Count (#)')) +
  theme_classic() +
  theme(axis.ticks=element_blank(), axis.text=element_blank(),
    axis.line=element_blank()) +
  coord_cartesian(
    xlim=buffered.range(grid.FPLY$long, 0.05),
    ylim=buffered.range(grid.FPLY$lat, 0.05)
  ) +
  xlab('') +
  ylab('') +
  ggtitle('Species richness')

```



**Figure 6** Species richness. Squares denote planning units. Planning units with a brighter color are inhabited by more species.

## Appendix S2: BayeScan Results

```
knitr::kable(
  format.table(
    ldply(
      seq_along(unique(spp.samples.DF$species)),
      function(i) {
        data.frame(
          Species=paste0('\\textit{',gsub('\\_', ' ', unique(spp.samples.DF$species)
          Individuals=nrow(spp.BayeScanData.LST[[i]]@matrix),
          Total_Loci=ncol(spp.BayeScanData.LST[[i]]@matrix),
          Neutral_Loci=sum(spp.BayeScan.LST[[i]]@results@fst[[7]]=='neutral'),
          Adpative_Loci=sum(spp.BayeScan.LST[[i]]@results@fst[[7]]=='adaptive'),
          Proportion_adaptive=sum(spp.BayeScan.LST[[i]]@results@fst[[7]]=='adaptive')
        )
      }
    ),
    digits=2,
    colnames=c('Species', 'Individuals (#)', 'Total Loci (#)', 'Neutral Loci (#)', 'Adaptive Loci (#)', 'Proportion adaptive'),
    align=c('l', 'c', 'c', 'c', 'c', 'c')
  )
)
```

Species	Individuals	Total_Loci	Neutral_Loci	Adpative_Loci	Proportion_adaptive
<i>Androsace obtusifolia</i>	131	138	66	72	0.52
<i>Arabis alpina</i>	385	151	52	99	0.66
<i>Campanula barbata</i>	307	114	46	68	0.60

```
knitr::kable(
  format.table(
    ldply(
      seq_along(unique(spp.samples.DF$species)),
      function(i) {
        data.frame(
          Species=paste0('\\textit{',gsub('\\_', ' ', unique(spp.samples.DF$species)
          Primer=spp.BayeScanData.LST[[i]]@primers,
          Probability=spp.BayeScan.LST[[i]]@results@fst[[2]],
          qval=spp.BayeScan.LST[[i]]@results@fst[[4]],
          alpha=spp.BayeScan.LST[[i]]@results@fst[[5]],
          fst=spp.BayeScan.LST[[i]]@results@fst[[6]],
          Type=spp.BayeScan.LST[[i]]@results@fst[[7]]
        )
      }
    )
  )
)
```

```

    ),
    omit='Type'
),
digits=2,
colnames=c('Species', 'Primer', 'Probability', 'q-value', '$\\alpha$', '$F_{ST}$', 'Type')
align=c('l', 'c', 'c', 'c', 'c', 'c', 'c')
)

```

Species	Primer	Probability	qval	alpha	fst	Type
<i>Androsace obtusifolia</i>	AAC_CAN_83.0	0.06	0.30	-0.02	0.09	neutral
	AAC_CAN_85.0	1.00	0.00	1.02	0.21	adaptive
	AAC_CAN_89.0	0.06	0.30	-0.01	0.09	neutral
	AAC_CAN_91.0	0.00	0.47	0.00	0.09	neutral
	AAC_CAN_100.0	0.39	0.06	-0.28	0.07	neutral
	AAC_CAN_102.0	0.24	0.09	-0.10	0.08	neutral
	AAC_CAN_108.0	0.04	0.35	-0.02	0.09	neutral
	AAC_CAN_124.0	1.00	0.00	-1.05	0.03	adaptive
	AAC_CAN_125.0	1.00	0.00	1.45	0.29	adaptive
	AAC_CAN_128.0	1.00	0.00	1.06	0.22	adaptive
	AAC_CAN_130.0	0.04	0.35	0.01	0.09	neutral
	AAC_CAN_132.0	0.94	0.01	-0.79	0.04	adaptive
	AAC_CAN_133.0	1.00	0.00	1.43	0.29	adaptive
	AAC_CAN_136.0	1.00	0.00	0.89	0.19	adaptive
	AAC_CAN_137.0	1.00	0.00	-2.31	0.01	adaptive
	AAC_CAN_146.0	0.00	0.47	0.00	0.09	neutral
	AAC_CAN_151.0	1.00	0.00	-1.01	0.04	adaptive
	AAC_CAN_152.0	1.00	0.00	-1.41	0.02	adaptive
	AAC_CAN_153.0	0.08	0.21	-0.02	0.09	neutral
	AAC_CAN_182.0	1.00	0.00	1.60	0.32	adaptive
	AAC_CAN_195.0	1.00	0.00	1.35	0.27	adaptive
	AAC_CAN_211.0	1.00	0.00	1.30	0.26	adaptive
	AAC_CAN_220.0	0.02	0.42	-0.01	0.09	neutral
	AAC_CAN_231.0	0.00	0.47	0.00	0.09	neutral
	AAC_CAN_239.0	0.96	0.00	-0.69	0.05	adaptive
	AAC_CAN_272.0	0.53	0.05	-0.16	0.08	adaptive
	AAC_CAN_319.0	1.00	0.00	-1.79	0.02	adaptive



Species	Primer	Probability	qval	alpha	fst	Type
	ACA_CAT_81.0	1.00	0.00	0.76	0.18	adaptive
	ACA_CAT_85.0	1.00	0.00	-0.87	0.04	adaptive
	ACA_CAT_90.0	1.00	0.00	-0.96	0.04	adaptive
	ACA_CAT_97.0	1.00	0.00	-0.87	0.04	adaptive
	ACA_CAT_99.0	1.00	0.00	-0.77	0.05	adaptive
	ACA_CAT_100.0	0.00	0.47	0.00	0.09	neutral
	ACA_CAT_102.0	0.02	0.42	0.01	0.09	neutral
	ACA_CAT_103.0	0.02	0.42	0.00	0.09	neutral
	ACA_CAT_108.0	0.02	0.42	0.01	0.09	neutral
	ACA_CAT_120.0	0.04	0.35	-0.01	0.09	neutral
	ACA_CAT_124.0	0.16	0.14	-0.05	0.09	neutral
	ACA_CAT_126.0	0.65	0.01	-0.46	0.06	adaptive
	ACA_CAT_129.0	0.02	0.42	-0.01	0.09	neutral
	ACA_CAT_131.0	0.16	0.14	0.05	0.09	neutral
	ACA_CAT_133.0	1.00	0.00	0.59	0.15	adaptive
	ACA_CAT_135.0	0.04	0.35	-0.02	0.09	neutral
	ACA_CAT_140.0	0.96	0.00	-0.78	0.04	adaptive
	ACA_CAT_148.0	1.00	0.00	-1.09	0.03	adaptive
	ACA_CAT_152.0	1.00	0.00	1.75	0.35	adaptive
	ACA_CAT_153.0	1.00	0.00	0.98	0.21	adaptive
	ACA_CAT_155.0	1.00	0.00	1.86	0.38	adaptive
	ACA_CAT_157.0	0.18	0.11	-0.07	0.08	neutral
	ACA_CAT_159.0	0.14	0.17	0.05	0.09	neutral
	ACA_CAT_162.0	1.00	0.00	-1.46	0.02	adaptive
	ACA_CAT_168.0	0.10	0.19	0.02	0.09	neutral
	ACA_CAT_173.0	0.96	0.00	-0.61	0.05	adaptive
	ACA_CAT_177.0	0.02	0.42	0.01	0.09	neutral
	ACA_CAT_178.0	0.06	0.30	-0.01	0.09	neutral
	ACA_CAT_187.0	1.00	0.00	0.97	0.21	adaptive
	ACA_CAT_192.0	0.06	0.30	0.00	0.09	neutral
	ACA_CAT_196.0	1.00	0.00	-1.18	0.03	adaptive
	ACA_CAT_199.0	0.02	0.42	-0.01	0.09	neutral

Species	Primer	Probability	qval	alpha	fst	Type
	ACA_CAT_200.0	0.04	0.35	-0.01	0.09	neutral
	ACA_CAT_204.0	1.00	0.00	-0.93	0.04	adaptive
	ACA_CAT_205.0	0.08	0.21	-0.03	0.09	neutral
	ACA_CAT_210.0	0.02	0.42	0.00	0.09	neutral
	ACA_CAT_214.0	1.00	0.00	1.07	0.22	adaptive
	ACA_CAT_219.0	0.06	0.30	0.00	0.09	neutral
	ACA_CAT_229.0	1.00	0.00	1.59	0.32	adaptive
	ACA_CAT_237.0	0.00	0.47	0.00	0.09	neutral
	ACA_CAT_243.0	1.00	0.00	-0.93	0.04	adaptive
	ACA_CAT_246.0	1.00	0.00	-1.54	0.02	adaptive
	ACA_CAT_248.0	1.00	0.00	-1.05	0.03	adaptive
	ACA_CAT_282.0	0.06	0.30	0.03	0.09	neutral
	ACA_CAT_300.0	1.00	0.00	-1.13	0.03	adaptive
	ACA_CAT_390.0	1.00	0.00	1.20	0.25	adaptive
	ACA_CAT_391.0	0.06	0.30	0.01	0.09	neutral
	ACA_CAT_393.0	0.31	0.08	-0.19	0.08	neutral
	AGG_CAA_82.3	0.94	0.01	-0.72	0.05	adaptive
	AGG_CAA_83.0	0.02	0.42	0.00	0.09	neutral
	AGG_CAA_84.2	1.00	0.00	-0.75	0.05	adaptive
	AGG_CAA_86.9	0.16	0.14	-0.12	0.08	neutral
	AGG_CAA_90.9	0.06	0.30	-0.02	0.09	neutral
	AGG_CAA_94.8	0.08	0.21	-0.04	0.09	neutral
	AGG_CAA_95.6	0.04	0.35	0.01	0.09	neutral
	AGG_CAA_100.0	0.61	0.03	-0.46	0.06	adaptive
	AGG_CAA_101.0	0.02	0.42	0.00	0.09	neutral
	AGG_CAA_109.3	1.00	0.00	-1.03	0.03	adaptive
	AGG_CAA_110.2	1.00	0.00	-1.01	0.04	adaptive
	AGG_CAA_113.5	1.00	0.00	-2.33	0.01	adaptive
	AGG_CAA_115.9	1.00	0.00	1.53	0.31	adaptive
	AGG_CAA_117.4	1.00	0.00	-1.36	0.03	adaptive
	AGG_CAA_118.2	0.06	0.30	-0.03	0.09	neutral
	AGG_CAA_122.9	0.63	0.03	-0.43	0.06	adaptive

Species	Primer	Probability	qval	alpha	fst	Type
	AGG_CAA_129.5	1.00	0.00	-1.61	0.02	adaptive
	AGG_CAA_130.1	0.06	0.30	0.03	0.09	neutral
	AGG_CAA_135.5	1.00	0.00	-1.10	0.03	adaptive
	AGG_CAA_137.9	0.06	0.30	-0.01	0.09	neutral
	AGG_CAA_144.9	0.00	0.47	0.00	0.09	neutral
	AGG_CAA_145.8	0.02	0.42	0.00	0.09	neutral
	AGG_CAA_150.9	0.02	0.42	-0.01	0.09	neutral
	AGG_CAA_152.8	1.00	0.00	-1.57	0.02	adaptive
	AGG_CAA_155.4	0.22	0.11	0.15	0.11	neutral
	AGG_CAA_164.0	0.22	0.11	-0.17	0.08	neutral
	AGG_CAA_175.9	1.00	0.00	-1.08	0.03	adaptive
	AGG_CAA_181.6	0.04	0.35	0.01	0.09	neutral
	AGG_CAA_182.1	0.98	0.00	-1.21	0.03	adaptive
	AGG_CAA_188.1	1.00	0.00	-1.30	0.03	adaptive
	AGG_CAA_191.3	0.37	0.07	-0.18	0.08	neutral
	AGG_CAA_193.9	0.04	0.35	-0.01	0.09	neutral
	AGG_CAA_196.8	0.43	0.06	-0.25	0.07	neutral
	AGG_CAA_201.7	0.12	0.18	0.01	0.09	neutral
	AGG_CAA_212.4	0.14	0.17	0.07	0.10	neutral
	AGG_CAA_215.6	1.00	0.00	-1.11	0.03	adaptive
	AGG_CAA_219.0	0.82	0.01	-0.63	0.05	adaptive
	AGG_CAA_225.2	1.00	0.00	-1.38	0.03	adaptive
	AGG_CAA_253.2	0.55	0.04	-0.41	0.07	adaptive
	AGG_CAA_256.7	0.53	0.05	-0.42	0.07	adaptive
	AGG_CAA_262.5	1.00	0.00	1.74	0.35	adaptive
	AGG_CAA_263.6	0.12	0.18	0.03	0.09	neutral
	AGG_CAA_264.8	1.00	0.00	-1.32	0.03	adaptive
	AGG_CAA_267.9	1.00	0.00	-1.13	0.03	adaptive
	AGG_CAA_269.8	1.00	0.00	-1.15	0.03	adaptive
	AGG_CAA_270.8	0.00	0.47	0.00	0.09	neutral
	AGG_CAA_276.8	0.94	0.01	-0.99	0.04	adaptive
	AGG_CAA_299.0	0.00	0.47	0.00	0.09	neutral

Species	Primer	Probability	qval	alpha	fst	Type
<i>Arabis alpina</i>	AGG_CAA_305.1	0.00	0.47	0.00	0.09	neutral
	AGG_CAA_312.1	1.00	0.00	-1.39	0.02	adaptive
	AGG_CAA_313.2	0.88	0.01	-0.93	0.04	adaptive
	AGG_CAA_316.0	0.63	0.03	0.68	0.18	adaptive
	AGG_CAA_319.0	0.14	0.17	-0.01	0.09	neutral
	AGG_CAA_324.1	0.02	0.42	0.00	0.09	neutral
	AGG_CAA_359.4	0.02	0.42	0.00	0.09	neutral
	AGG_CAA_360.5	0.06	0.30	0.01	0.09	neutral
	AGG_CAA_363.2	0.00	0.47	0.00	0.09	neutral
	AGG_CAA_364.1	1.00	0.00	-1.01	0.04	adaptive
	AGG_CAA_376.2	1.00	0.00	-1.15	0.03	adaptive
	AGG_CAA_376.7	0.00	0.47	0.00	0.09	neutral
	AGG_CAA_396.0	1.00	0.00	-0.90	0.04	adaptive
	AGG_CAA_403.4	0.06	0.30	-0.01	0.09	neutral
	AGG_CAA_420.5	1.00	0.00	-1.78	0.02	adaptive
	AAT_CAC_51.9	0.04	0.14	-0.01	0.13	neutral
	AAT_CAC_54.7	1.00	0.00	0.96	0.28	adaptive
	AAT_CAC_69.0	1.00	0.00	0.25	0.17	adaptive
	AAT_CAC_73.6	1.00	0.00	-0.58	0.08	adaptive
	AAT_CAC_77.9	1.00	0.00	-1.69	0.03	adaptive
	AAT_CAC_86.9	1.00	0.00	-1.66	0.03	adaptive
	AAT_CAC_90.5	0.27	0.04	0.22	0.17	neutral
	AAT_CAC_95.3	1.00	0.00	-0.63	0.08	adaptive
	AAT_CAC_97.0	0.04	0.14	0.01	0.14	neutral
	AAT_CAC_97.1	0.00	0.33	0.00	0.13	neutral
	AAT_CAC_100.3	0.02	0.19	0.00	0.13	neutral
	AAT_CAC_105.4	1.00	0.00	-2.00	0.02	adaptive
	AAT_CAC_118.4	1.00	0.00	-0.53	0.09	adaptive
	AAT_CAC_121.3	0.00	0.33	0.00	0.13	neutral
	AAT_CAC_128.0	1.00	0.00	-1.12	0.05	adaptive
	AAT_CAC_130.0	1.00	0.00	-0.97	0.06	adaptive
	AAT_CAC_147.4	0.08	0.07	-0.01	0.13	neutral

Species	Primer	Probability	qval	alpha	fst	Type
	AAT_CAC_156.9	0.00	0.33	0.00	0.13	neutral
	AAT_CAC_175.1	1.00	0.00	-0.68	0.07	adaptive
	AAT_CAC_177.9	0.00	0.33	0.00	0.13	neutral
	AAT_CAC_179.6	1.00	0.00	-2.15	0.02	adaptive
	AAT_CAC_188.6	1.00	0.00	0.92	0.28	adaptive
	AAT_CAC_190.0	1.00	0.00	-1.52	0.03	adaptive
	AAT_CAC_195.5	1.00	0.00	1.57	0.42	adaptive
	AAT_CAC_197.1	1.00	0.00	1.73	0.46	adaptive
	AAT_CAC_200.6	1.00	0.00	-1.17	0.05	adaptive
	AAT_CAC_201.8	1.00	0.00	-1.22	0.05	adaptive
	AAT_CAC_209.2	0.04	0.14	-0.01	0.13	neutral
	AAT_CAC_213.1	1.00	0.00	0.78	0.25	adaptive
	AAT_CAC_215.0	0.35	0.02	0.09	0.15	neutral
	AAT_CAC_216.2	0.00	0.33	0.00	0.13	neutral
	AAT_CAC_217.0	0.00	0.33	0.00	0.13	neutral
	AAT_CAC_218.1	0.00	0.33	0.00	0.13	neutral
	AAT_CAC_219.1	1.00	0.00	-1.21	0.04	adaptive
	AAT_CAC_225.5	1.00	0.00	-0.82	0.06	adaptive
	AAT_CAC_227.1	1.00	0.00	-1.04	0.05	adaptive
	AAT_CAC_229.1	0.00	0.33	0.00	0.13	neutral
	AAT_CAC_231.4	1.00	0.00	1.60	0.43	adaptive
	AAT_CAC_249.7	1.00	0.00	-1.54	0.03	adaptive
	AAT_CAC_259.3	1.00	0.00	-1.68	0.03	adaptive
	AAT_CAC_277.3	0.00	0.33	0.00	0.13	neutral
	AAT_CAC_298.1	0.27	0.04	-0.04	0.13	neutral
	AAT_CAC_315.9	1.00	0.00	1.21	0.34	adaptive
	AAT_CAC_330.4	1.00	0.00	-1.69	0.03	adaptive
	AAT_CAC_334.8	0.02	0.19	0.00	0.13	neutral
	AAT_CAC_336.6	1.00	0.00	0.50	0.20	adaptive
	AAT_CAC_353.0	0.31	0.03	-0.19	0.12	neutral
	AAT_CAC_359.2	1.00	0.00	-1.59	0.03	adaptive
	AAT_CAC_399.9	0.00	0.33	0.00	0.13	neutral

Species	Primer	Probability	qval	alpha	fst	Type
	AAT_CAC_410.5	1.00	0.00	-1.96	0.02	adaptive
	AAT_CAC_412.4	1.00	0.00	-2.07	0.02	adaptive
	AAT_CAC_458.1	1.00	0.00	-1.83	0.02	adaptive
	AAT_CAC_488.6	1.00	0.00	1.57	0.42	adaptive
	AGT_CAC_53.8	1.00	0.00	-1.97	0.02	adaptive
	AGT_CAC_56.3	1.00	0.00	-1.86	0.02	adaptive
	AGT_CAC_98.2	1.00	0.00	-0.73	0.07	adaptive
	AGT_CAC_104.9	1.00	0.00	-1.11	0.05	adaptive
	AGT_CAC_114.8	0.04	0.14	0.00	0.13	neutral
	AGT_CAC_145.8	0.84	0.00	0.37	0.18	adaptive
	AGT_CAC_154.2	0.00	0.33	0.00	0.13	neutral
	AGT_CAC_158.1	1.00	0.00	-0.88	0.06	adaptive
	AGT_CAC_169.0	1.00	0.00	-1.27	0.04	adaptive
	AGT_CAC_171.1	0.02	0.19	0.00	0.13	neutral
	AGT_CAC_181.1	1.00	0.00	1.30	0.36	adaptive
	AGT_CAC_183.0	1.00	0.00	0.35	0.18	adaptive
	AGT_CAC_184.4	0.02	0.19	0.00	0.13	neutral
	AGT_CAC_191.2	1.00	0.00	-2.10	0.02	adaptive
	AGT_CAC_195.0	1.00	0.00	0.97	0.29	adaptive
	AGT_CAC_200.9	0.02	0.19	0.00	0.13	neutral
	AGT_CAC_203.8	1.00	0.00	1.48	0.40	adaptive
	AGT_CAC_205.8	0.10	0.06	0.03	0.14	neutral
	AGT_CAC_210.0	1.00	0.00	-0.70	0.07	adaptive
	AGT_CAC_230.8	1.00	0.00	0.78	0.25	adaptive
	AGT_CAC_241.0	0.00	0.33	0.00	0.13	neutral
	AGT_CAC_245.6	1.00	0.00	0.91	0.27	adaptive
	AGT_CAC_264.7	1.00	0.00	-1.26	0.04	adaptive
	AGT_CAC_266.9	1.00	0.00	-2.56	0.01	adaptive
	AGT_CAC_269.4	0.00	0.33	0.00	0.13	neutral
	AGT_CAC_274.0	1.00	0.00	1.39	0.38	adaptive
	AGT_CAC_285.6	1.00	0.00	1.35	0.37	adaptive
	AGT_CAC_291.5	1.00	0.00	1.56	0.42	adaptive

Species	Primer	Probability	qval	alpha	fst	Type
	AGT_CAC_295.6	1.00	0.00	-2.22	0.02	adaptive
	AGT_CAC_315.2	0.04	0.14	0.01	0.14	neutral
	AGT_CAC_332.1	0.00	0.33	0.00	0.13	neutral
	AGT_CAC_347.8	1.00	0.00	1.81	0.48	adaptive
	AGT_CAC_355.2	1.00	0.00	-1.92	0.02	adaptive
	AGT_CAC_360.2	0.00	0.33	0.00	0.13	neutral
	AGT_CAC_386.5	1.00	0.00	1.15	0.32	adaptive
	AGT_CAC_418.5	0.00	0.33	0.00	0.13	neutral
	AGT_CAC_420.2	0.00	0.33	0.00	0.13	neutral
	AGT_CAC_444.3	1.00	0.00	1.27	0.35	adaptive
	AGT_CAC_453.4	1.00	0.00	-0.51	0.09	adaptive
	AGT_CAC_458.5	1.00	0.00	-2.03	0.02	adaptive
	AGT_CAC_489.1	1.00	0.00	1.55	0.42	adaptive
	ATC_CAC_52.4	0.04	0.14	0.00	0.13	neutral
	ATC_CAC_56.7	0.00	0.33	0.00	0.13	neutral
	ATC_CAC_61.5	1.00	0.00	-0.83	0.06	adaptive
	ATC_CAC_64.3	0.24	0.05	-0.06	0.13	neutral
	ATC_CAC_91.9	0.00	0.33	0.00	0.13	neutral
	ATC_CAC_96.2	1.00	0.00	1.70	0.45	adaptive
	ATC_CAC_99.5	1.00	0.00	0.57	0.21	adaptive
	ATC_CAC_100.6	1.00	0.00	-1.20	0.05	adaptive
	ATC_CAC_102.0	1.00	0.00	-1.28	0.04	adaptive
	ATC_CAC_111.9	0.47	0.01	-0.17	0.12	neutral
	ATC_CAC_113.5	1.00	0.00	-2.05	0.02	adaptive
	ATC_CAC_123.5	1.00	0.00	1.75	0.46	adaptive
	ATC_CAC_139.7	1.00	0.00	1.62	0.43	adaptive
	ATC_CAC_140.8	1.00	0.00	1.30	0.36	adaptive
	ATC_CAC_142.9	1.00	0.00	-1.37	0.04	adaptive
	ATC_CAC_144.1	0.16	0.05	-0.05	0.13	neutral
	ATC_CAC_148.6	1.00	0.00	1.98	0.52	adaptive
	ATC_CAC_149.8	1.00	0.00	1.40	0.38	adaptive
	ATC_CAC_151.8	0.04	0.14	0.01	0.13	neutral

Species	Primer	Probability	qval	alpha	fst	Type
	ATC_CAC_156.1	1.00	0.00	1.15	0.32	adaptive
	ATC_CAC_162.5	0.00	0.33	0.00	0.13	neutral
	ATC_CAC_181.9	1.00	0.00	-1.49	0.03	adaptive
	ATC_CAC_186.4	1.00	0.00	-1.69	0.03	adaptive
	ATC_CAC_189.9	0.04	0.14	-0.01	0.13	neutral
	ATC_CAC_194.8	0.00	0.33	0.00	0.13	neutral
	ATC_CAC_198.3	1.00	0.00	-1.18	0.05	adaptive
	ATC_CAC_199.4	0.00	0.33	0.00	0.13	neutral
	ATC_CAC_204.2	0.84	0.00	0.55	0.21	adaptive
	ATC_CAC_207.3	0.02	0.19	0.00	0.13	neutral
	ATC_CAC_215.8	0.96	0.00	-0.49	0.09	adaptive
	ATC_CAC_220.7	1.00	0.00	-0.73	0.07	adaptive
	ATC_CAC_223.7	1.00	0.00	-1.35	0.04	adaptive
	ATC_CAC_229.1	1.00	0.00	-1.35	0.04	adaptive
	ATC_CAC_230.7	1.00	0.00	0.76	0.25	adaptive
	ATC_CAC_233.7	1.00	0.00	0.69	0.24	adaptive
	ATC_CAC_235.8	1.00	0.00	-0.54	0.08	adaptive
	ATC_CAC_258.7	0.00	0.33	0.00	0.13	neutral
	ATC_CAC_266.2	1.00	0.00	1.53	0.41	adaptive
	ATC_CAC_270.8	0.06	0.08	-0.02	0.13	neutral
	ATC_CAC_273.6	1.00	0.00	-0.94	0.06	adaptive
	ATC_CAC_274.9	1.00	0.00	1.58	0.42	adaptive
	ATC_CAC_276.4	0.00	0.33	0.00	0.13	neutral
	ATC_CAC_277.8	1.00	0.00	-1.00	0.05	adaptive
	ATC_CAC_287.8	1.00	0.00	1.49	0.40	adaptive
	ATC_CAC_288.7	1.00	0.00	1.30	0.36	adaptive
	ATC_CAC_332.4	1.00	0.00	-1.36	0.04	adaptive
	ATC_CAC_347.9	1.00	0.00	1.81	0.48	adaptive
	ATC_CAC_370.4	1.00	0.00	1.42	0.38	adaptive
	ATC_CAC_373.3	0.47	0.01	-0.19	0.11	neutral
	ATC_CAC_378.0	1.00	0.00	-1.63	0.03	adaptive
	ATC_CAC_387.7	0.00	0.33	0.00	0.13	neutral



Species	Primer	Probability	qval	alpha	fst	Type
<i>Campanula barbata</i>	ATC_CAC_401.5	1.00	0.00	-1.93	0.02	adaptive
	ATC_CAC_405.7	1.00	0.00	1.56	0.42	adaptive
	ATC_CAC_430.5	0.02	0.19	0.00	0.13	neutral
	ATC_CAC_442.2	1.00	0.00	1.57	0.42	adaptive
	ATC_CAC_445.2	1.00	0.00	1.13	0.32	adaptive
	ATC_CAC_456.3	0.02	0.19	0.00	0.13	neutral
	ACA_CTA_55.8	1.00	0.00	1.97	0.46	adaptive
	ACA_CTA_69.2	1.00	0.00	1.87	0.43	adaptive
	ACA_CTA_101.6	0.47	0.02	-0.22	0.09	neutral
	ACA_CTA_114.7	0.14	0.07	0.05	0.12	neutral
	ACA_CTA_122.8	0.06	0.13	0.01	0.11	neutral
	ACA_CTA_132.9	1.00	0.00	-1.37	0.03	adaptive
	ACA_CTA_153.4	1.00	0.00	-0.92	0.05	adaptive
	ACA_CTA_155.4	0.00	0.38	0.00	0.11	neutral
	ACA_CTA_164.8	0.00	0.38	0.00	0.11	neutral
	ACA_CTA_169.4	1.00	0.00	-0.71	0.06	adaptive
	ACA_CTA_174.3	0.00	0.38	0.00	0.11	neutral
	ACA_CTA_178.0	1.00	0.00	2.01	0.47	adaptive
	ACA_CTA_179.5	0.43	0.03	-0.17	0.10	neutral
	ACA_CTA_183.6	0.06	0.13	0.00	0.11	neutral
	ACA_CTA_186.8	0.67	0.01	0.30	0.15	adaptive
	ACA_CTA_187.9	0.04	0.17	0.00	0.11	neutral
	ACA_CTA_194.5	1.00	0.00	-1.66	0.02	adaptive
	ACA_CTA_195.9	0.00	0.38	0.00	0.11	neutral
	ACA_CTA_197.7	1.00	0.00	-0.81	0.05	adaptive
	ACA_CTA_203.8	1.00	0.00	-0.54	0.07	adaptive
	ACA_CTA_213.5	1.00	0.00	-1.00	0.04	adaptive
	ACA_CTA_254.3	1.00	0.00	-0.69	0.06	adaptive
	ACA_CTA_284.5	1.00	0.00	-1.01	0.04	adaptive
	ACA_CTA_289.0	0.00	0.38	0.00	0.11	neutral
	ACA_CTA_296.4	0.00	0.38	0.00	0.11	neutral
	ACA_CTA_311.1	1.00	0.00	1.34	0.32	adaptive

Species	Primer	Probability	qval	alpha	fst	Type
	ACA_CTA_347.7	1.00	0.00	-1.32	0.03	adaptive
	ACA_CTA_368.6	0.00	0.38	0.00	0.11	neutral
	ACA_CTA_378.8	0.00	0.38	0.00	0.11	neutral
	ACA_CTA_382.7	0.31	0.05	-0.11	0.10	neutral
	ACA_CTA_393.5	1.00	0.00	-1.64	0.02	adaptive
	ACA_CTA_415.5	1.00	0.00	-1.13	0.04	adaptive
	ACA_CTA_489.5	1.00	0.00	-1.43	0.03	adaptive
	ACA_CTA_491.1	1.00	0.00	-2.29	0.01	adaptive
	AGA_CAC_84.2	1.00	0.00	-1.47	0.03	adaptive
	AGA_CAC_89.4	0.00	0.38	0.00	0.11	neutral
	AGA_CAC_91.6	1.00	0.00	1.73	0.40	adaptive
	AGA_CAC_102.6	1.00	0.00	-0.66	0.06	adaptive
	AGA_CAC_106.2	0.04	0.17	0.01	0.11	neutral
	AGA_CAC_107.0	0.18	0.06	0.06	0.12	neutral
	AGA_CAC_109.0	1.00	0.00	1.25	0.30	adaptive
	AGA_CAC_116.9	1.00	0.00	-0.73	0.06	adaptive
	AGA_CAC_130.4	1.00	0.00	0.76	0.21	adaptive
	AGA_CAC_133.3	0.02	0.25	0.00	0.11	neutral
	AGA_CAC_135.7	1.00	0.00	1.71	0.40	adaptive
	AGA_CAC_141.4	0.00	0.38	0.00	0.11	neutral
	AGA_CAC_168.9	1.00	0.00	1.03	0.25	adaptive
	AGA_CAC_170.2	0.04	0.17	-0.01	0.11	neutral
	AGA_CAC_180.6	1.00	0.00	-0.75	0.06	adaptive
	AGA_CAC_183.6	1.00	0.00	-1.33	0.03	adaptive
	AGA_CAC_196.1	0.00	0.38	0.00	0.11	neutral
	AGA_CAC_202.0	0.06	0.13	0.02	0.11	neutral
	AGA_CAC_204.8	0.00	0.38	0.00	0.11	neutral
	AGA_CAC_214.4	0.00	0.38	0.00	0.11	neutral
	AGA_CAC_218.3	0.02	0.25	0.01	0.11	neutral
	AGA_CAC_227.1	1.00	0.00	-1.08	0.04	adaptive
	AGA_CAC_231.4	0.02	0.25	0.01	0.11	neutral
	AGA_CAC_245.4	0.00	0.38	0.00	0.11	neutral

Species	Primer	Probability	qval	alpha	fst	Type
	AGA_CAC_247.3	1.00	0.00	-1.68	0.02	adaptive
	AGA_CAC_250.2	1.00	0.00	-1.16	0.04	adaptive
	AGA_CAC_251.1	0.00	0.38	0.00	0.11	neutral
	AGA_CAC_269.0	0.02	0.25	0.01	0.11	neutral
	AGA_CAC_279.6	1.00	0.00	-1.56	0.03	adaptive
	AGA_CAC_283.4	0.04	0.17	-0.01	0.11	neutral
	AGA_CAC_285.4	1.00	0.00	-0.58	0.07	adaptive
	AGA_CAC_286.8	0.41	0.04	-0.28	0.09	neutral
	AGA_CAC_294.8	0.39	0.04	-0.38	0.09	neutral
	AGA_CAC_299.4	1.00	0.00	-1.37	0.03	adaptive
	AGA_CAC_308.2	1.00	0.00	0.82	0.22	adaptive
	AGA_CAC_314.3	1.00	0.00	0.64	0.19	adaptive
	AGA_CAC_316.2	1.00	0.00	-1.11	0.04	adaptive
	AGA_CAC_318.3	1.00	0.00	-1.65	0.02	adaptive
	AGA_CAC_321.0	1.00	0.00	1.25	0.30	adaptive
	AGA_CAC_324.2	1.00	0.00	-1.58	0.03	adaptive
	AGA_CAC_326.2	1.00	0.00	-1.43	0.03	adaptive
	AGA_CAC_338.4	1.00	0.00	-0.40	0.08	adaptive
	AGA_CAC_356.5	1.00	0.00	-1.44	0.03	adaptive
	AGA_CAC_441.7	0.02	0.25	0.00	0.11	neutral
	AGA_CAC_445.1	0.00	0.38	0.00	0.11	neutral
	AGA_CAC_475.7	1.00	0.00	-0.89	0.05	adaptive
	AGA_CAC_477.9	0.00	0.38	0.00	0.11	neutral
	AGA_CAC_487.7	1.00	0.00	-0.79	0.05	adaptive
	AGT_CTG_85.2	1.00	0.00	-0.63	0.06	adaptive
	AGT_CTG_109.9	1.00	0.00	-2.03	0.02	adaptive
	AGT_CTG_127.9	0.02	0.25	0.00	0.11	neutral
	AGT_CTG_130.9	0.08	0.09	0.02	0.11	neutral
	AGT_CTG_135.5	1.00	0.00	0.62	0.19	adaptive
	AGT_CTG_144.4	1.00	0.00	1.01	0.25	adaptive
	AGT_CTG_151.4	0.02	0.25	-0.01	0.11	neutral
	AGT_CTG_152.8	0.94	0.00	0.60	0.19	adaptive

Species	Primer	Probability	qval	alpha	fst	Type
	AGT_CTG_181.2	1.00	0.00	-1.24	0.04	adaptive
	AGT_CTG_191.2	1.00	0.00	0.78	0.21	adaptive
	AGT_CTG_196.4	1.00	0.00	-0.97	0.05	adaptive
	AGT_CTG_202.3	0.02	0.25	0.00	0.11	neutral
	AGT_CTG_218.5	1.00	0.00	-1.36	0.03	adaptive
	AGT_CTG_221.0	1.00	0.00	1.57	0.37	adaptive
	AGT_CTG_226.4	1.00	0.00	-0.60	0.06	adaptive
	AGT_CTG_228.9	0.55	0.01	-0.23	0.09	adaptive
	AGT_CTG_230.8	1.00	0.00	-2.07	0.02	adaptive
	AGT_CTG_234.9	0.02	0.25	0.00	0.11	neutral
	AGT_CTG_245.4	1.00	0.00	-1.88	0.02	adaptive
	AGT_CTG_262.1	1.00	0.00	-0.54	0.07	adaptive
	AGT_CTG_266.2	0.00	0.38	0.00	0.11	neutral
	AGT_CTG_297.5	0.12	0.08	-0.02	0.11	neutral
	AGT_CTG_336.5	1.00	0.00	-0.98	0.05	adaptive
	AGT_CTG_344.2	1.00	0.00	-1.49	0.03	adaptive
	AGT_CTG_359.0	0.02	0.25	0.00	0.11	neutral
	AGT_CTG_363.4	0.00	0.38	0.00	0.11	neutral
	AGT_CTG_392.1	1.00	0.00	-1.62	0.02	adaptive
	AGT_CTG_415.3	1.00	0.00	-0.74	0.06	adaptive
	AGT_CTG_443.5	1.00	0.00	-2.02	0.02	adaptive
	AGT_CTG_452.7	0.00	0.38	0.00	0.11	neutral
	AGT_CTG_459.7	1.00	0.00	-1.18	0.04	adaptive
	AGT_CTG_488.9	1.00	0.00	-1.71	0.02	adaptive

## Appendix S3: Genomic MDS

```
knitr::kable(
  format.table(
    ldply(
      seq_along(unique(spp.samples.DF$species)),
      function(i) {
        ldply(
          seq_along(spp.mds.LST[[i]]),
```

```

        function(j) {
        data.frame(
            Species=paste0('\\textit{',gsub('\\_',' ', unique(spp.samples.DF$species),
            Loci=names(spp.mds.LST[[i]])[j],
            Stress=spp.mds.LST[[i]][[j]]$stress,
            Converged=spp.mds.LST[[i]][[j]]$converged
        )
        })
    }
),
    omit='Converged'
),
digits=2,
caption='Summary of non-metric multi-dimensional scaling (MDS) analyses on genetic variation',
colnames=c('Species', 'Loci Type', 'NMDS Stress', 'Converged'),
align=c('l', 'c', 'c', 'c')
)

```

Species	Loci	Stress	Converged
<i>Androsace obtusifolia</i>	adaptive	0.24	FALSE
	neutral	0.22	FALSE
<i>Arabis alpina</i>	adaptive	0.25	FALSE
	neutral	0.21	FALSE
<i>Campanula barbata</i>	adaptive	0.20	FALSE
	neutral	0.23	FALSE

**Table 3** Summary of non-metric multi-dimensional scaling (MDS) analyses on genetic variation for each species.

## Appendix S4: Distribution maps of intra-specific variation

```

plot.spp.mds <- function(i) {
  do.call(
    grid.arrange,
    append(
      unlist(llply(c('adaptive','neutral'), function(g) {
        llply(seq_len(mds.k), function(k) {
          ggplot() +
            geom_polygon(data=countries.FPLY, aes(x=long, y=lat, group=group),
              fill='grey20', color='grey80') +
            geom_polygon(data=grid.FPLY, aes_string(x='long', y='lat',

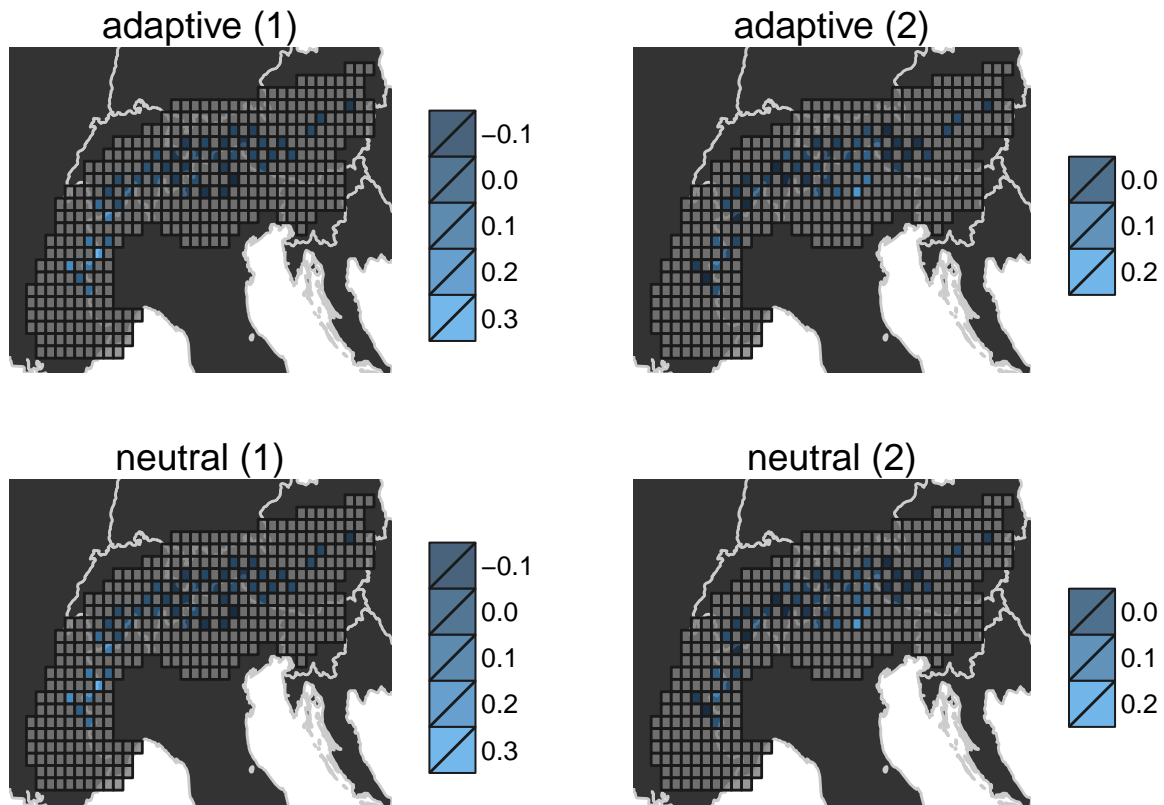
```

```

        group='group', fill=paste0(unique(spp.samples.DF$species)[i], '_'),
        alpha=0.8, color='grey10') +
        guides(fill=guide_legend(title=' ')) +
        theme_classic() +
        theme(axis.ticks=element_blank(), axis.text=element_blank(),
              plot.margin=unit(c(0,0,0,0),'cm'), axis.line=element_blank()) +
        coord_cartesian(
          xlim=buffered.range(grid.FPLY$long, 0.05),
          ylim=buffered.range(grid.FPLY$lat, 0.05)
        ) +
        xlab('') +
        ylab('') +
        ggtitle(paste0(g, ' (',k,')'))
      })
    },recursive=FALSE),
    list(ncol=2)
  )
}

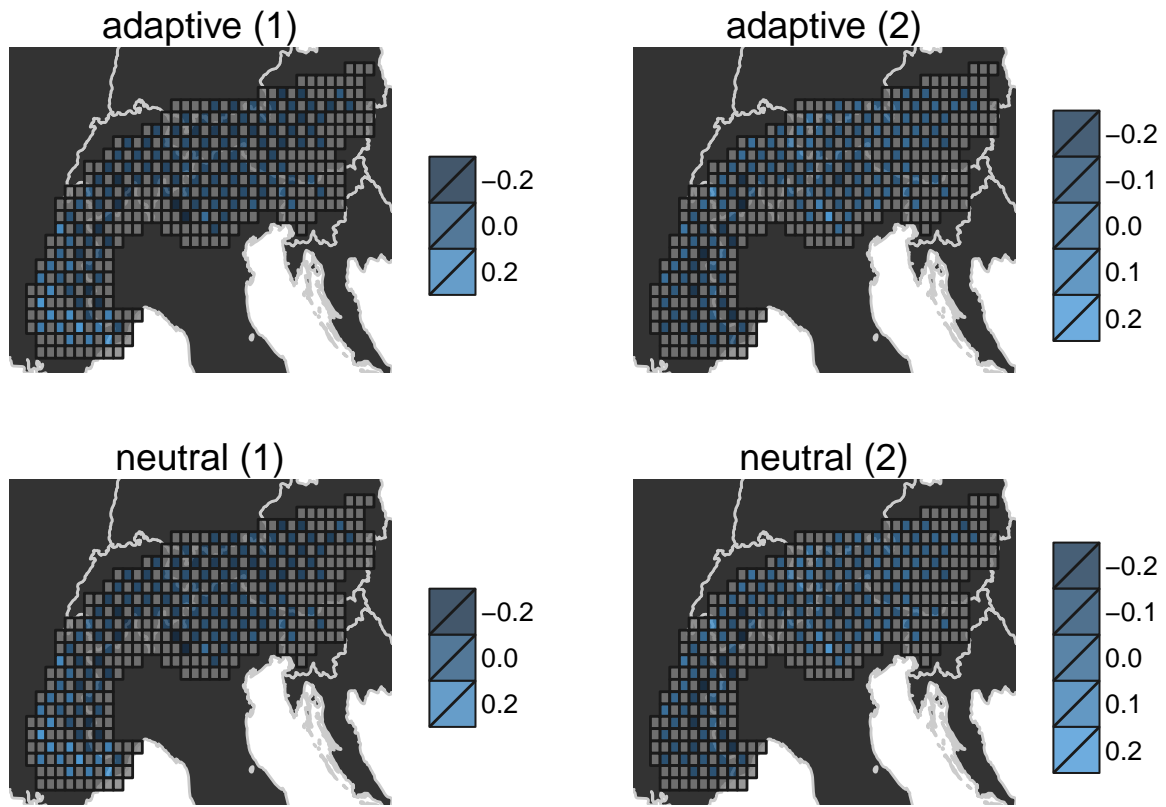
```

```
plot.spp.mds(1)
```



**Figure 7** Distribution of adaptive and neutral genetic variation in *Androsace obtusifolia*. Each square represents a planning unit. The color of each planning unit panel corresponds to ordination values. Planning units with similar colors contain individuals with similar genetic variation.

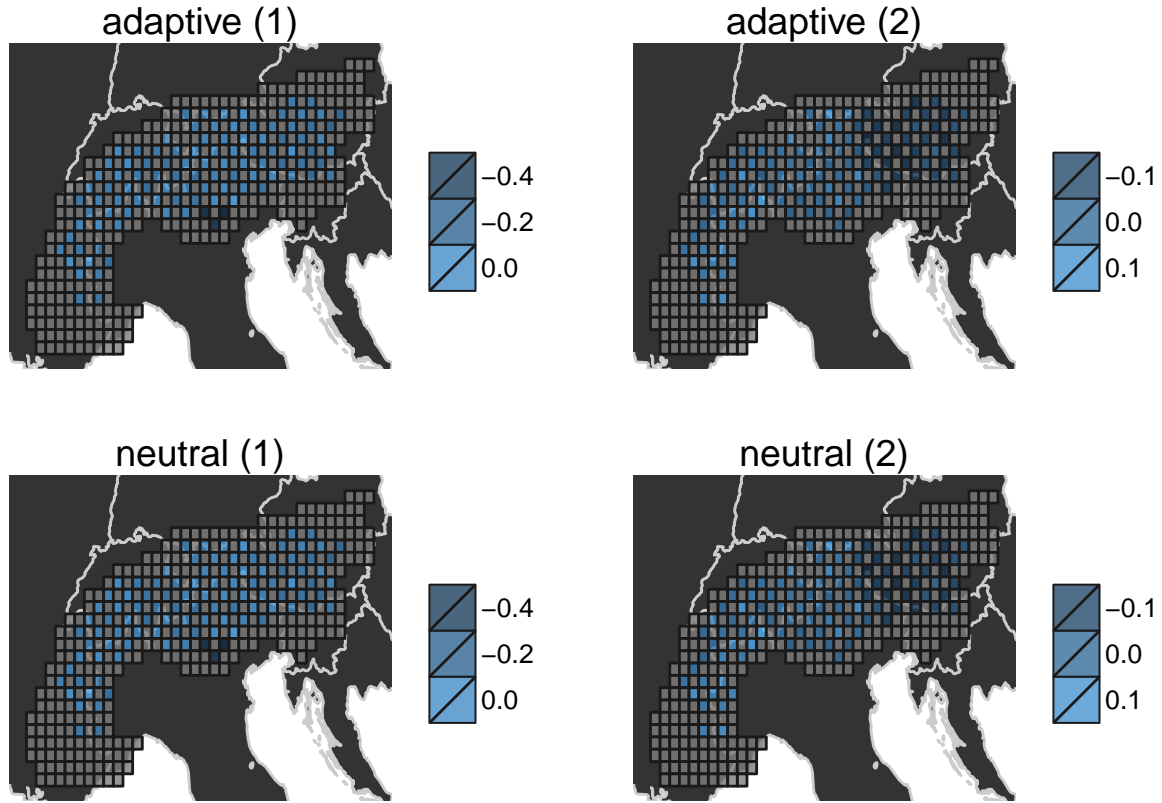
```
plot.spp.mds(2)
```



**Figure 8** Distribution of adaptive and neutral genetic variation in *Arabis alpina*. See Figure XX caption for conventions.

`plot.spp.mds(3)`





**Figure 9** Distribution of adaptive and neutral genetic variation in *Campanula barbata*. See Figure XX caption for conventions.

## Appendix S5: Principle components analysis on climatic variation

```
## load pca summary
pca.DF <- read.table('extdata/BioClim_variables/pca.TXT', skip=80) %>% 'names<-'(
  c('Principle Component', 'Eigen Value', 'Variation explained (%)',
    'Accumulative variation explained (%)')
)
## make results table showing Eigen values
knitr::kable(
  pca.DF,
  digits=2,
  caption='Summary of priciniple components analysis (PCA) on bioclimatic variation across t
  align=c('l', 'c', 'c', 'c')
)
```

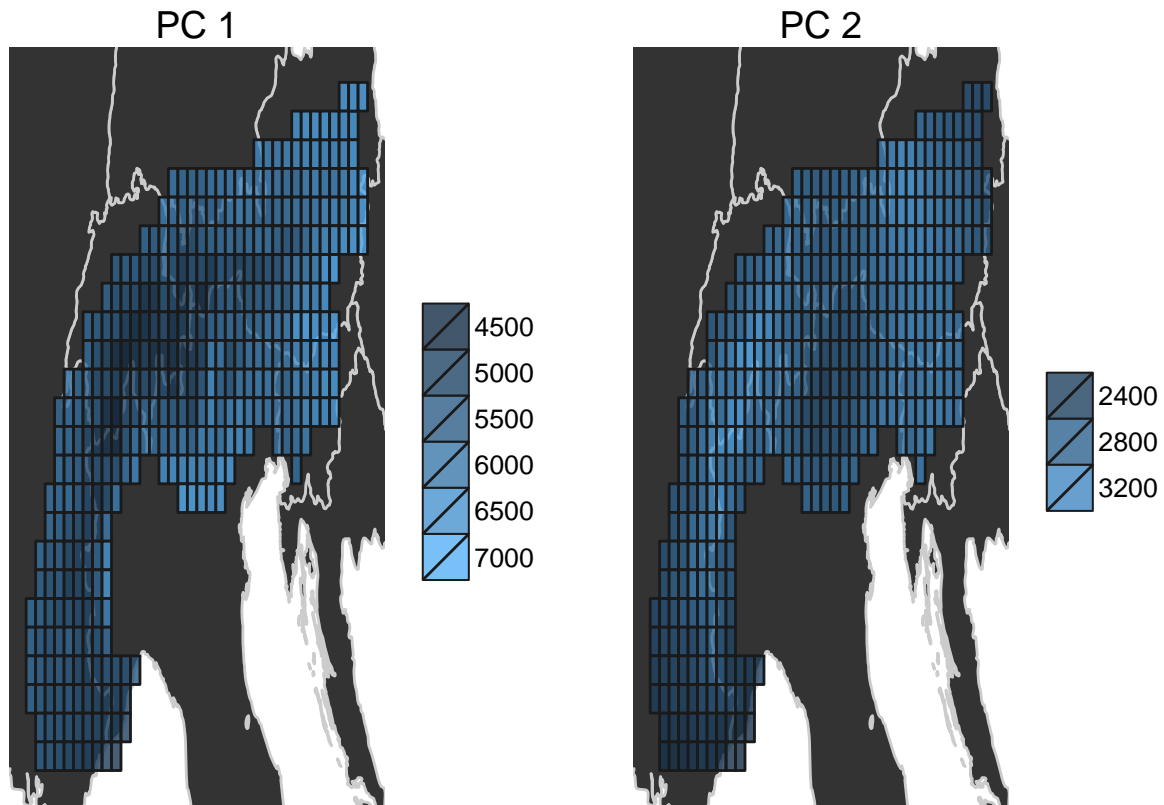
Principle Component	Eigen Value	Variation explained (%)	Accumulative variation explained (%)
1	216765.14	82.67	82.67

Principle Component	Eigen Value	Variation explained (%)	Accumulative variation explained (%)
2	38177.84	14.56	97.23
3	5356.75	2.04	99.27
4	1216.67	0.46	99.73
5	700.39	0.27	100.00

**Table 4** Summary of priciniple components analysis (PCA) on bioclimatic variation across the study area. The first two principle components (PCs) were used for subsequent analysis.

## Appendix S6: Maps of climatic variation

```
do.call(
  grid.arrange,
  append(
    lapply(grep('^env\\_.*$', names(grid.DF), value=TRUE), function(x) {
      ggplot() +
        geom_polygon(data=countries.FPLY, aes(x=long, y=lat, group=group),
          fill='grey20', color='grey80') +
        geom_polygon(data=grid.FPLY, aes_string(x='long', y='lat',
          group='group', fill=x),
          alpha=0.8, color='grey10') +
        guides(fill=guide_legend(title=' ')) +
        theme_classic() +
        theme(axis.ticks=element_blank(), axis.text=element_blank(),
          plot.margin=unit(c(0,0,0,0),'cm'), axis.line=element_blank()) +
        coord_cartesian(
          xlim=buffered.range(grid.FPLY$long, 0.05),
          ylim=buffered.range(grid.FPLY$lat, 0.05)
        ) +
        xlab('') +
        ylab('') +
        ggtitle(paste0('PC ', substr(x, nchar(x), nchar(x))))
    }),
    list(ncol=2)
  )
)
```



**Figure 10** Climatic variation. Each panel depicts variation based on a different principle component (PC). Squares represent planning units. The color of each planning unit denotes the average principle component value of pixels inside it. Planning units with more similar colors have more similar climate regimes.

Meirmans, P., Goudet, J., IntraBioDiv Consortium, Gaggiotti, O. (2011) Ecology and life history affect different aspects of the population structure of 27 high-alpine plants. *Molecular Ecology*. **20**, 3144–3155.