Модели пространственного распределения видов

Автор: Владимир Шитиков
https://stok1946.blogspot.com/

Изучение структуры пространственного распределения экологических сообществ и его связей с условиями обитания различных популяций является важнейшим направлением биосферных исследований. После появления в 1980-х годах пакета BIOCLIM (Busby, 1991) моделирование распределения видов (SDM – *Species Distribution Models*) и экологических ниш (ENM – *Environmental Niche Models*) стало мощным инструментом (макро)-экологических и биогеографических исследований и оценки роли факторов, влияющих на распространение видов (Peterson et al., 2011). Эти методы оказались также весьма эффективными в палеоэкологии, филогенетике, управлении биоресурсами и охране дикой природы (Araújo et al. 2019). Появилось огромное количество литературы по различным методам SDM / ENM, использование которых широко освещено в работах зарубежных экологов (Franklin, 2009; Guisan et al., 2017) и подробном обзоре наших коллег из МГУ (Лисовский и др., 2020).

Анализ пространственного распределения видов основан на двух различных концептуальных подходах (Ovaskainen, Abrego, 2020). Процессно-ориентированные SDM (также известные как ранговые модели динамики популяций – Zurell et al. 2016) включают в явной форме модельные структуры и параметры, описывающие механизмы основных экологических процессов в сообществах. Необходимость оценки коэффициентов интенсивности размножения, смертности, расселения и демографической стохастичности (Vellend, 2016), а также их зависимость от выборочных процессов получения данных, делают такой подход пока еще труднодоступным, хотя учет базовых процессов в сообществах должен приветствоваться в любых случаях (D'Amen et al. 2017).

Другой подход можно назвать коррелятивным, в том смысле, что он основан на нахождении статистических зависимостей между факторами окружающей среды и данными о встречаемости видов. Описаны десятки методов построения SDM (Norberg et al., 2019), которые различаются такими аспектами, как состав исходных данных ("только присутствие" видов в точках отбора проб, "присутствие-отсутствие" или количественная оценка обилия), структурные допущения моделей (обобщенная линейная модель, опорные векторы или случайный лес), алгоритмы получения решения (использование максимума правдоподобия или байесовский подход) и техническая реализация (доступен ли метод в виде R-пакета или как самостоятельный программный продукт). Успешно ведутся работы по ранжированию совокупности построенных моделей по степени их компетентности и построению ансамблей (коллективов), в которых предсказания нескольких моделей взвешиваются и усредняются (Breiner et al. 2015).

При всем множестве опубликованных работ до сих пор в полной мере отсутствует не только единая теория, но и конкретные практические рекомендации построения SDM. Это обусловлено как объективно существующим многообразием изучаемых экологических сообществ, природно-климатических зон, жизненных форм и техник проведения наблюдений, так и большим арсеналом разработанных методов компьютерной обработки и верификации моделей, выбор которых в значительной мере определяется субъективными взглядами исследователей. В частности, развернутый анализ результатов использования 33 моделей SDM на сообществах птиц, бабочек, деревьев и травянистой растительности показал (Norberg, 2019), что успех моделирования зависит от типа полученных данных на 36%, постановки задачи

(интерполяция или экстраполяция) на 26%, выбранного алгоритма на 33% и объема выборки – только на 2%.

Ниже рассматривается построение моделей SDM различными методами с использованием гидробиологических данных по обилию бентосных организмов в малых реках крупного региона.

1. Подготовка исходных данных

Модели распределения видов будем строить на примере данных гидробиологической съемки донных сообществ бассейна Средней и Нижней Волги (Зинченко, 2011) в разные месяцы вегетационного периода 1990-2019 гг. Гидробиологическую съемку макрозообентоса проводили на 90 малых и 12 средних равнинных реках, притоках Куйбышевского, Саратовского и Волгоградского водохранилищ, в том числе, на 6 реках аридного региона бассейна оз. Эльтон. Всего было выделено S = 740 видов и таксонов бентоса рангом выше вида.

В нашем предыдущем сообщении «Интерполяция и визуализация пространственных данных» мы уже подробно останавливались на этом исследовании и сформировали компьютерную карту региона для визуализации результатов интерполяции. Объект ggplot2, воспроизводящий эту карту, представлен в файле по адресу: http://www.ievbras.ru/ecostat/Kiril/R/Blog/WB_map.RData.

В другом файле на этом общедоступном ресурсе http://www.ievbras.ru/ecostat/Kiril/R/Blog/WB_data.RData приведены точечные данные, привязанные к географическим координатам 132 участков рек, где выполнялись гидробиологические пробы. В таблице `df` объединены значения некоторых показателей биоразнообразия и обилия видов, в том числе, число видов в пробе `N_Spec`, индекс Шеннона `Shennon`, индекс `EPT`, численность различных таксономических групп бентоса `ORTHOCLADIINAE`, `PRODIAMESINAE` или некоторых видов *Procladius ferrugineus* – `ChPrc.f.` и *Cricotopus* gr. *Sylvestris* – `ChCri.s.` Таблица `df_var` содержит наблюдаемые гидрохимические показатели в тех же точках отбора проб – минерализация, насыщение кислородом `O2`, содержание ионов аммония `NH4` и категория донного грунта `Ground` (от 1 – песок или галька до 6 – черные илы). Чтобы выполнить некоторые последующие скрипты, необходимо поместить скачанные файлы `WB_map.RData` и `WB_data.RData` в рабочий каталог среды R.

Кроме точечных данных из таблицы `df_var` в качестве предикторов для построения моделей SDM будем использовать биоклиматические и геофизические показатели в виде сеток (или матриц *grid*), представленные на общедоступных серверах. Определим предварительно географический экстент (т.е. прямоугольник области проведения исследования) по крайним точкам на севере, юге, западе и востоке:

```
load(file="WB_data.RData")
obs.data <- df[, c("X", "Y")]
# определим крайние координаты точек
max.lat <- ceiling(max(obs.data$Y))
min.lat <- floor(min(obs.data$Y))
max.lon <- ceiling(max(obs.data$X))
min.lon <- floor(min(obs.data$X))
library("sp")
library("raster")
library("rgdal")
geographic.extent <- extent(x = c(min.lon, max.lon, min.lat, max.lat))
# к этому экстенту добавим поля до совпадения с картой из WB_map.RData
geographic.extent <- extent(x = c(44.532, 55.648, 48.655, 55.112))
```

Слои биоклиматических данных мировой базы WorldClim (Hijmans et al., 2005) по сетке географических координат с различным ее разрешением включают 19 показателей, которые покрывают глобальные площади суши, за исключением Антарктиды. Они кодируются следующим образом:

BIO1 = среднегодовая температура
BIO2 = средний суточный диапазон (максимальная температура - минимальная температура))
BIO3 = изотермичность (BIO2/BIO7) (×100)
BIO4 = сезонность температуры (стандартное отклонение ×100)
BIO5 = максимальная температура самого теплого месяца
BIO6 = минимальная температура самого холодного месяца
BIO7 = годовой диапазон температур (BIO5-BIO6)
BIO8 = средняя температура самой влажной четверти
BIO9 = средняя температура самого сухого квартала
BIO10 = средняя температура самого теплого квартала
BIO11 = средняя температура самой холодной четверти
BIO12 = годовое количество осадков
BIO13 = осадки самого влажного месяца
BIO14 = дождь в засушливый месяц
BIO15 = сезонность осадков (коэффициент вариации)
BIO16 = осадки самого влажного квартала
BIO17 = осадков в сухой четверти
BIO18 = осадки самого теплого квартала
BIO19 = осадки самого холодного квартала

В специальном разделе alt базы данных представлена также высота над уровнем моря (м).

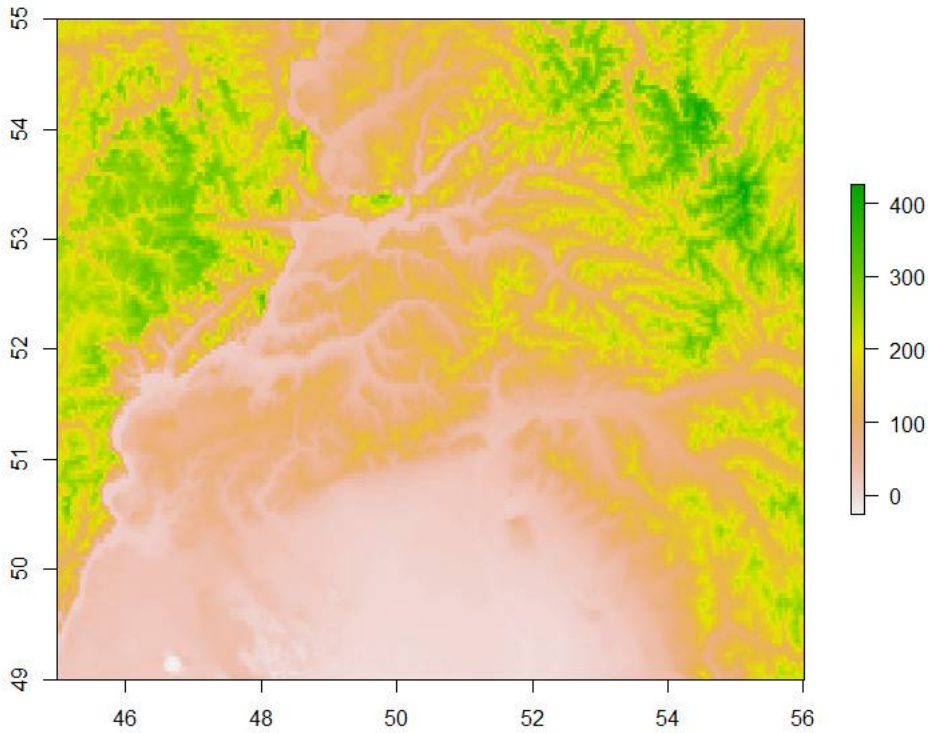
Обратим внимание, что данные о температуре указаны в целочисленном формате °C·10, что позволяет значительно уменьшить размеры файлов. Это означает, что значение 231 представляет собой 23,1 °C. Данные об осадках представлены в миллиметрах (мм).

```
#  Загрузка с сервера WorldClim данных разрешением 2.5 минуты
bioclim.data <- getData(name = "worldclim", var = "bio",
                                  res = 2.5, path = "")
alt.data <- getData(name = "worldclim", var = "alt",
          res = 2.5, path = "")
 Предупреждения:
1: В .newCRS(value) :
  +proj=longlat +datum=WGS84 +no_defs is not a valid PROJ.4 CRS string
2: В .newCRS(value) :
  +proj=longlat +datum=WGS84 is not a valid PROJ.4 CRS string
```

Предупреждения указывают, что мы не задали код нужной нам географической проекции. Параллельно с таблицами данных в оперативной памяти в рабочем каталоге R будет создана папка wc2-5 с архивными файлами этих данных, которые можно использовать для локальной работы без доступа в интернет.

В памяти компьютера образовались два специальных геоинформационных объекта типа растр (*raster*) – многоканальный (точнее, 20-канальный) bioclim.data и одноканальный alt.data, каждый из которых содержит слои с сетками данных. Пакет raster обладает большим набором функций работы с этими объектами. Вначале обрежем растры биоклиматических показателей и высоты по выделенному географическому экстенту:

```
bioclim.data <- crop(x = bioclim.data, y = geographic.extent)
alt.data <- crop(x = alt.data, y = geographic.extent)
plot(alt.data)
```



Добавим к анализу еще один показатель - индекс шероховатости рельефа (*Terrain ruggedness index* - TRI), т.е. топографический индекс, показывающий среднее значение перепада высот между анализируемой ячейкой и восемью соседними. Этот индекс можно рассчитать с использованием функции `tri()`. Объединим все данные в один многоканальный растр (стек растров), который содержит по каждому из 21 показателей матрицу данных из 155*267 = 41385 ячеек:

```
library(spatialEco)
tri.app <- tri(alt.data, exact = FALSE)
ClimAlt.data <- addLayer(bioclim.data,alt.data)
ClimAltTri.data <- addLayer(ClimAlt.data, tri.app)
names(ClimAltTri.data)[21] <- 'tri'
class       : RasterStack
dimensions : 155, 267, 41385, 21  (nrow, ncol, ncell, nlayers)
resolution : 0.04166667, 0.04166667  (x, y)
extent      : 44.54167, 55.66667, 48.66667, 55.125  (xmin, xmax, ymin, ymax)
crs         : NA
names      :       bio1,       bio2,       bio3,       bio4,       bio5, ...
min values :         22,         80,         19,      11098,        239, ...
max values :    86.0000,   116.0000,    23.0000, 13444.0000,   324.0000, ...
```

На следующем этапе нам необходимо получить значения биоклиматических показателей для каждой из 132 географических точек, где проводилось взятие гидробиологических проб.

```
library("dismo")
# Выгрузка в таблицу значений растра в точках гидробиологических проб
bc.model <- bioclim(x = ClimAltTri.data, p = obs.data)
df.clim <- as.data.frame(bc.model@presence)
```

Естественно, что все биоклиматические данные представляют сильно коррелированный набор переменных. Чтобы избежать эффекта коллинеарности при построении моделей, выберем несколько базовых переменных с минимальным уровнем фактора инфляции дисперсии (VIF):

```
library(car)
y_lm <- log(df$ChPrc.f. + 1)  # В качестве отклика - Procladius ferrugineus
m <- lm(y_lm ~ .,data=df.clim)
summary(m)
m=update(m, .~.- bio7)  # Удаляем переменную с коэффициентом  NA
vif(m)
       bio1        bio2        bio3        bio4        bio5        bio6
1573.642667  164.107714   20.262554  189.271175 1052.534507 1148.867939
    bio8        bio9       bio10       bio11       bio12
3.402921    7.707940 1307.821942 2010.766538 2119.884518
      bio13       bio14       bio15       bio16       bio17       bio18
267.937030  166.608827   48.433324  476.811507  387.477497 1749.883900
      bio19         alt       tri
 376.932891   15.486714    2.062036
```

Переменную `BIO7` сразу пришлось удалить, поскольку она связана детерминированной зависимостью с другими показателями. Ограничимся в дальнейших расчетах пятью климатическими факторами (трех температурных и двух по количеству осадков), высотой и шероховатостью рельефа (`tri`).

```
ClimAltTri7.data <- dropLayer(ClimAltTri.data, c(2,4:7,9:14,16,18:19))
class      : RasterStack
dimensions : 155, 267, 41385, 7  (nrow, ncol, ncell, nlayers)
resolution : 0.04166667, 0.04166667  (x, y)
extent     : 44.54167, 55.66667, 48.66667, 55.125  (xmin, xmax, ymin, ymax)
crs        : NA
names      :     bio1,     bio3,     bio8,    bio15,    bio17,      alt,      tri
min values :       22,       19,      -80,       14,       42,      -26,        0
max values :  86.0000,  23.0000, 232.0000,  38.0000, 118.0000, 426.0000, 384.0378
```

Для проведения последующих операций тестирования моделей выгрузим содержимое семиканального растра вместе с географическими координатам в обычную таблицу данных. Обратим внимание, что значения шероховатости рельефа по краям географической области не могут быть корректно определены и равны NA. Восстановим пропущенные значения с помощью функции `preProcess()` из пакета `caret`. Сохраним сформированные объекты в файле для дальнейшего использования:

```
dfR <- cbind(coordinates(ClimAltTri7.data), values(ClimAltTri7.data))
ind <- apply(dfR, 1, function(x) sum(is.na(x))) > 0
nrow(dfR[ind,])
[1] 840
library(caret)
pPmI <- preProcess(as.data.frame(dfR[, 9]), method = 'medianImpute')
Imp <- as.vector(predict(pPmI, as.data.frame(dfR[, 9])))
dfR[, 9] <- Imp$"dfR[, 9]"
dfR <- as.data.frame(dfR)
head(dfR)
             x        y bio1 bio3 bio8 bio15 bio17 alt      tri
 [1,] 44.56250 55.10417   38   20  179    28    90 174 54.05553
 [2,] 44.60417 55.10417   38   20  179    28    91 179 54.05553
 [3,] 44.64583 55.10417   38   20  179    28    90 172 54.05553
 [4,] 44.68750 55.10417   38   20  179    29    89 171 54.05553
 [5,] 44.72917 55.10417   38   20  179    28    90 171 54.05553
 [6,] 44.77083 55.10417   39   20  179    29    88 164 54.05553
save(ClimAltTri7.data, ClimAltTri.data, df.clim, dfR, file="BioClim.RData")
```

2. Модель BIOCLIM, не использующая данные об отсутствии вида

В экологических исследованиях для многих организмов или ситуаций достоверно может быть определено только наличие (*only 'presence' data*) того или иного вида в точках проведения наблюдений, тогда как "ареалы отсутствия" часто выделить затруднительно, поскольку данных об отсутствии вида (*'absence' data*) либо нет, либо они являются предвзятыми и неполными. Разумеется, если исследователь располагает полным комплектом надежных эмпирических данных о присутствии и отсутствии, он может и должен использовать классический подход к их анализу, основанный на логистической регрессии или распознавании образов. Но если данных об отсутствии нет, использовать такие методы нельзя и это обусловило разработку различных алгоритмов, основанных на многократном случайном выборе подмножества точек, где, как предполагается, вид отсутствует (*pseudo-absence*, или "*background*" *points*).

Фоновые данные ("*background*" *points* - Phillips et al. 2006) не пытаются напрямую "угадать" точки отсутствия вида, а скорее характеризуют окружающую среду в исследуемом регионе. При этом анализ данных о присутствии должен установить, в каких условиях вид является более устойчивым и будет присутствовать с большей вероятностью, чем в среднем. Тесно связанное с фоном, но несколько отличное по смыслу понятие "*pseudo-absence*", которое также используется для генерации класса 0 при построении логистических моделей, пытается угадать точки отсутствия.

Модели распределения видов с учетом этих ограничений могут быть построены разными по сложности методами. Наиболее простые из них, так называемые методы экологических конвертов, к которым относится BIOCLIM (Busby, 1991), ограничивают искомую область распространения граничными значениями включенных в анализ факторов. Рассмотрим использование этого алгоритма фоновых точек на основе функций пакета `dismo`. Выполним формирование распределения прогнозируемой вероятности появления вида *Procladius ferrugineus.*

```
library("dismo")
# Построение модели распределения вида ChPrc.f.
obs.Yes <- df[df$ChPrc.f. !=0, (4:5)]
nrow(obs.Yes)
[1] 93
geographic.extent <- extent(x = c(44.532, 55.648, 48.655, 55.112))
bc.model <- bioclim(x = ClimAltTri7.data, p = obs.Yes)
```

С помощью функции `predict()` из пакета `dismo` построим искомое распределение вида и отобразим его на карте вместе с точками, где он наблюдался (обозначены ниже кружками с заливкой).

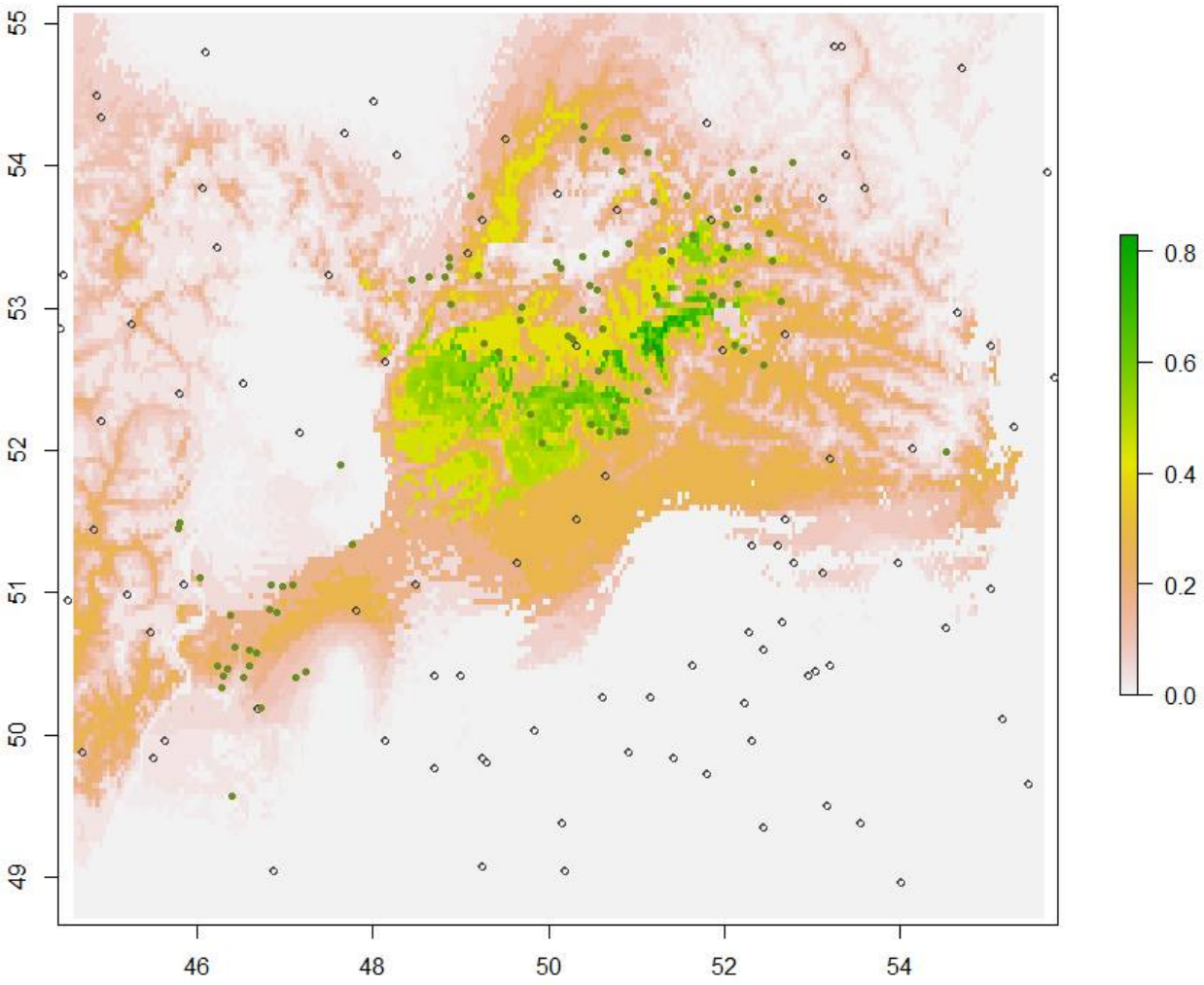
```
predict.presence <- dismo::predict(object = bc.model,
     x = ClimAltTri7.data,
     ext = geographic.extent)
plot(predict.presence)
points(obs.Yes$X, obs.Yes$Y, col = "olivedrab", pch = 20, cex = 0.95)
```

Сформируем и добавим на карту набор случайных фоновых точек в том же количестве, что и точек присутствия, т.е. 93.

```
# Создание набора случайных фоновых точек
# (в том же количестве, что и точек присутствия)
background <- randomPoints(mask = ClimAltTri7.data, # Разрешение из растра
     n = nrow(obs.Yes),                              # Число случайных точек
     ext = geographic.extent,          # Пространственное ограничение точек
     extf = 1)                     # В том же самом географическом экстенте
```

Добавляем на карту фоновые точки (обозначены кружками без заливки).

```
points(background, col = "grey30", pch = 1, cex = 0.85)
```



Оценим теперь, насколько верна наша модель. Для этого разобьем наборы наблюдаемых и фоновых точек на 5 групп каждый: четыре группы попеременно используется для построения модели, а по пятой будем проводить ее тестирование.

```
# Создание вектора разбиений на группы
group.presence <- kfold(x = obs.Yes, k = 5) # kfold - функция пакета dismo
head(group.presence)
[1] 4 2 3 3 3 4
# Проверим число точек в каждой группе
table(group.presence)
group.presence
 1  2  3  4  5
19 18 19 18 19
group.background <- kfold(x = background, k = 5)
spp <- 0
spa <- 0
sAUC <- 0
# Каждую группу поочередно выделяем в качестве тестовой
for (testing.group in 1:5) {
# Делим наблюдения на обучающую и тестовую последовательности
   presence.train <- obs.Yes[group.presence != testing.group, ]
   presence.test <- obs.Yes[group.presence == testing.group, ]
```

```
# Повторяем эту процедуру для pseudo-absence точек
   background.train <- background[group.background != testing.group, ]
   background.test <- background[group.background == testing.group, ]
# Построение модели распределения на обучающих данных
   bc.model <- bioclim(x = ClimAltTri7.data, p = presence.train)
# Используем тестовые данные для оценки модели
   bc.eval <- evaluate(p = presence.test,    # Тестовые данные присутствия
                  a = background.test,        # Тестовые данные отсутствия
                     model = bc.model,                # Оцениваемая модель
                 x = ClimAltTri7.data)       # Растр переменных для модели
# Определение минимального порога для "presence"
   bc.threshold <- threshold(x = bc.eval, stat = "spec_sens")
   spp <- spp + sum(bc.eval@presence > bc.threshold)
   spa <- spa + sum(bc.eval@absence < bc.threshold)
   sAUC <- sAUC + bc.eval@auc
}
c(spp, spp/nrow(obs.Yes)) # Правильное угадывание присутствия
[1] 71.  0.7634409
c(spa, spa/nrow(obs.Yes))  # Правильное угадывание отсутствия
[1] 70.  0.7526882
sAUC/5
[1] 0.7910831
```

Функция `evaluate()` предоставляет большой набор критериев для оценки качества модели. Мы ограничились AUC (англ. *area under ROC curve*, *площадь под ROC-кривой*). Чем выше показатель AUC, тем качественнее классификатор, при этом значение 0,5 демонстрирует непригодность выбранного метода классификации (соответствует случайному гаданию). Значение AUC = 0.79 соответствует школьной оценке примерно «четыре с минусом».

Функция `threshold()` представляет собой ряд средств определения порогового значения вероятности, выше которой вид считается обнаруженным. Параметр `stat = "spec_sens"` заставляет функцию устанавливать порог, при котором сумма чувствительности (истинный положительный уровень прогноза) и специфичности (истинный отрицательный уровень прогноза) является самой высокой.

Не слишком большой процент правильно оцененных случаев наличия вида (76.34%) свидетельствует о том, что этот факт далеко не всегда зависит от семи биоклиматических показателей, которые использовались при построении модели. Доля ошибок для фоновых точек определяется как несовершенством модели, так и тем, что в данной точке действительно возможно появление этого вида.

3. Метод максимальной энтропии MaxEnt

В последние десятилетия наблюдается возрастающая роль применения теории информации в экологии; например, для прогнозирования численности вида по функциональным признакам или других макроэкологических закономерностей (Harte, 2011). Для моделирования распределения видов был разработан метод максимальной энтропии, реализованный в программе MaxEnt (Phillips et al., 2006; Лисовский и др., 2020a) и получивший широкое распространение. С байесовской точки зрения принцип максимальной энтропии утверждает, что из всех возможных распределений вероятностей при известных ограничениях, распределение с наибольшей энтропией наилучшим образом представляет данные. Как и BIOCLIM, MaxEnt предсказывает вероятность присутствия вида в произвольной точке географического пространства, основываясь только на точках, где он уже был зарегистрирован (*presence-only*).

Концептуально Maxent сравнивает данные о наблюдаемом присутствии вида ($\gamma = 1$) с вектором z экологических предикторов, описывающим условия окружающей среды в изучаемом регионе. Если определить $f(z)$ как плотность вероятности

предикторов по всему региону и $f1(z)$ как плотность вероятности ковариат в точках того же региона, где встречается данный вид, то MaxEnt выполняет анализ отношения $f1(z)/ f(z)$. Алгоритм оптимизации использует предикторы из наблюдаемой и фоновой выборок и ищет такое распределение $f1(z)$, которое имеет максимальное расстояние от $f(z)$, т.е. максимальную относительную энтропию $f1(z)$ по отношению к $f(z)$. Действительно, $f(z)$ рассматривается здесь как нулевая модель для $f1(z)$, поскольку нет никаких оснований ожидать, что вид предпочтет какие-либо конкретные условия окружающей среды в отсутствие данных о его встречаемости. В результате наилучшим прогнозом являются такие условия окружающей среды, которые пропорциональны популяционной плотности вида в регионе.

Итогом работы MaxEnt является расчет экспоненциальной функции, аргументами которой являются частные функции отдельных предикторов (линейные, квадратичные, множественные и др.) с коэффициентами λ, оценивающими вклад соответствующего экологического фактора. Пошаговый выбор оптимальной модели и настройка коэффициентов λ осуществляется с учетом минимизации ошибки предсказания как на исходной выборке *presence-only*, так и на множестве случайно отобранных точек *background points*, где, как предполагается, вид отсутствует.

Для выполнения расчетов может быть использована функция `maxent()` из пакета `dismo`, но эта функция является лишь "R-оберткой" к java-программе, которую можно скачать с доступных ресурсов. Скачанный файл 'maxent.jar' необходимо положить в папку 'java' этого пакета.

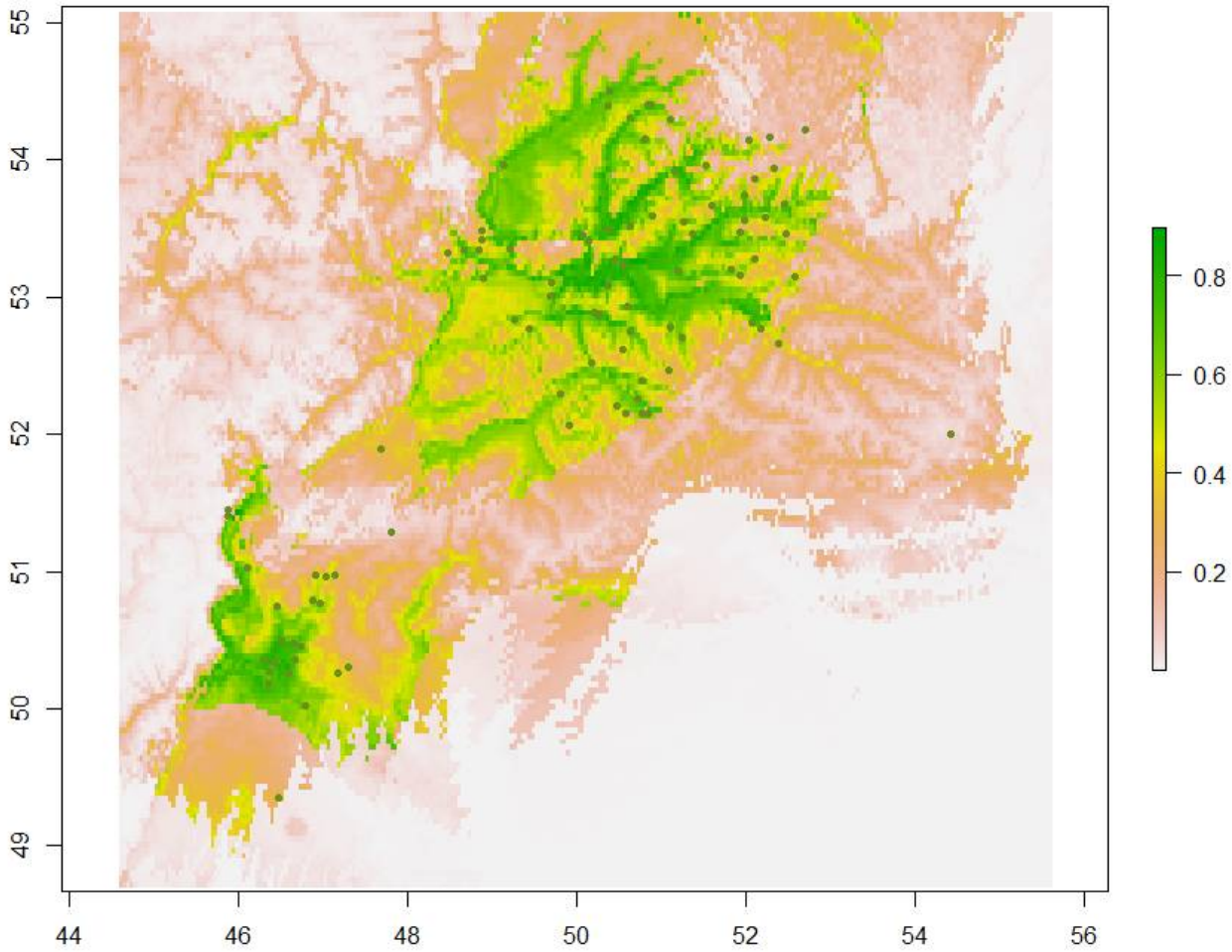
```
jar <- paste(system.file(package="dismo"), "/java/maxent.jar", sep='')
if (!(file.exists(jar) & require(rJava))) {
   print ("Please download it here:
http://www.cs.princeton.edu/~schapire/maxent/")
}
```

Расчеты будем выполнять почти по той же схеме и с использованием тех же исходных данных, что и для метода BIOCLIM (раздел 2). Выполним вначале плдгонку модели:

```
me.model <- maxent(x = ClimAltTri7.data, p = obs.Yes)
```

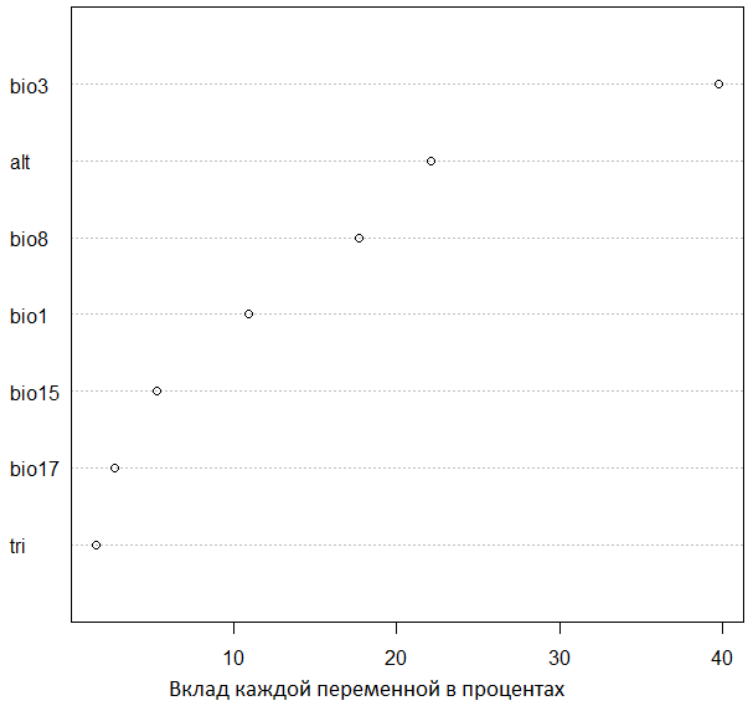
С помощью функции `predict()` из пакета `dismo` построим искомое распределение вида *Procladius ferrugineus* и отобразим его на карте вместе с точками, где он наблюдался (обозначены ниже кружками с заливкой). Отметим, что область с экологической пригодностью выше 0.6 занимает более обширную площадь и теснее охватывает эмпирические данные, чем это имело место на аналогичном графике в разделе 2.

```
# Создадим растр с вероятностями с использованием тех же данных
me.pred.presence <- predict(me.model,  ClimAltTri7.data)
# график "Прогнозируемая экологическая пригодность"
plot(me.pred.presence)
points(obs.Yes$X, obs.Yes$Y, col = "olivedrab", pch = 20, cex = 0.95)
```

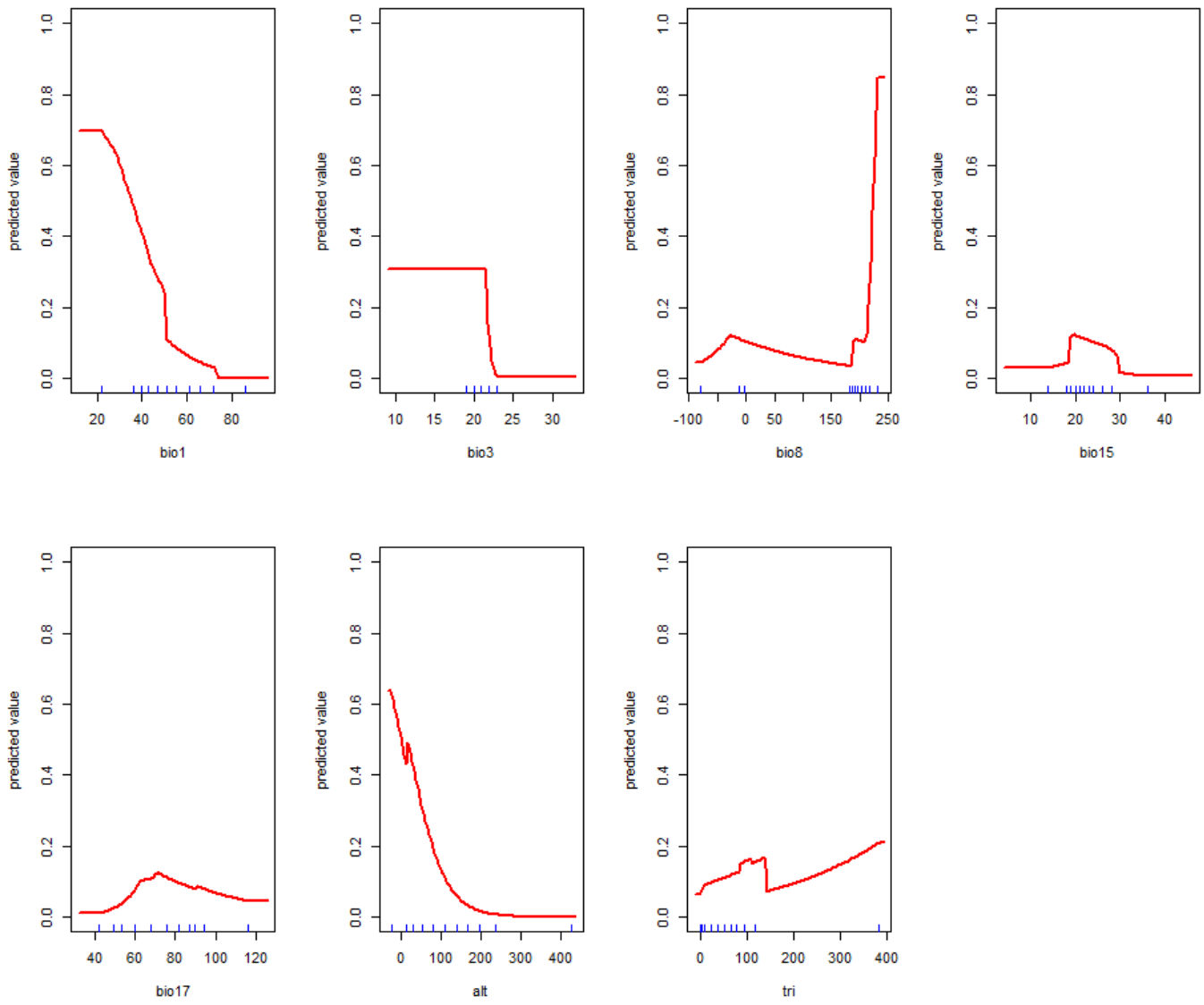


Два графика позволяют оценить относительную важность (вклад) биоклиматических показателей при построении модели и кривые отклика для каждого из них

```
# график, показывающий важность каждой переменной
plot(me.model)
```



```
# кривые отклика
response(me.model)
```

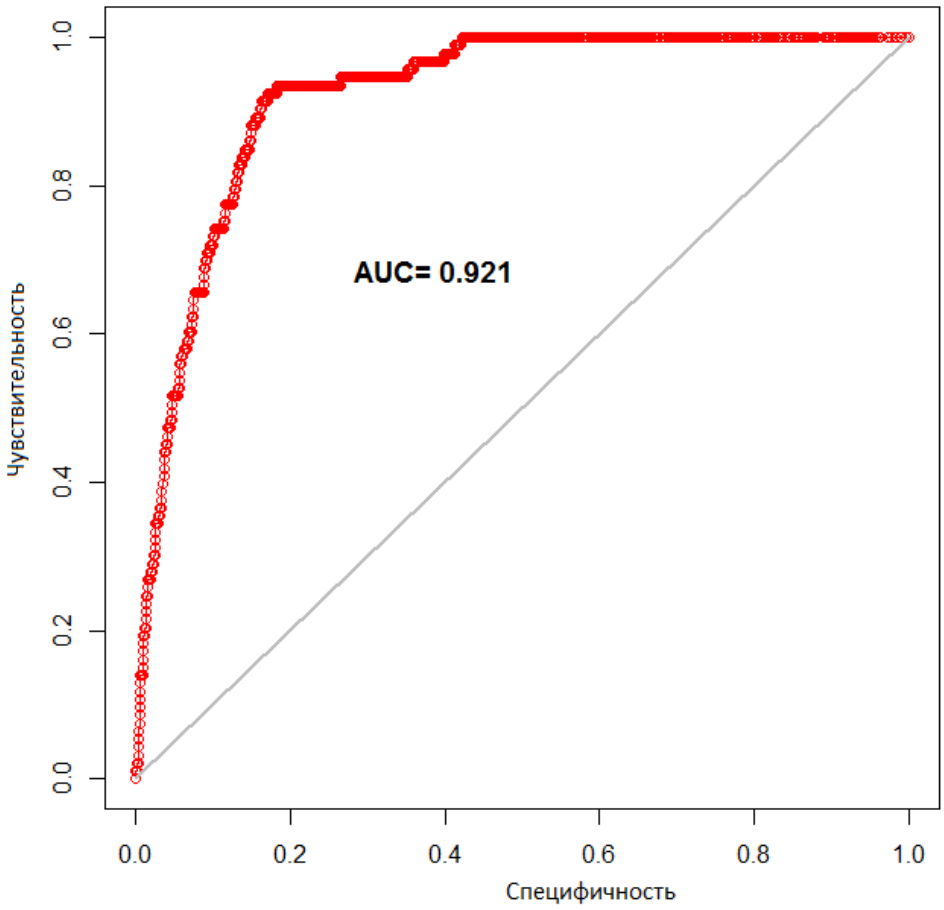


Графики показывают, что важнейшими экогеографическими факторами появления вида являются низкие значения среднегодовой температуры (bio1), изотермичности (bio3) и высоты над уровнем моря (alt).

Как и в предыдущем разделе, качество построенной модели можно оценить по величине AUC и доле правильных распознаваний. Это можно сделать как на полной выборке, так и в режиме кросс-проверки. Для этого надо предварительно создать случайный набор фоновых точек (пусть их будет 1000, хотя при подгонке модели по умолчанию использовалось 10000 фоновых точек).

```
# создание набора фоновых точек
bg <- randomPoints(me.pred.presence, 1000)

# Тестирование модели
# простейший путь - использование функции 'evaluate'
e1 <- evaluate(me.model, p=obs.Yes, a=bg, x=ClimAltTri7.data)
plot(e1, 'ROC')
```



Более непредвзятой оценкой качества модели является кросс-проверка. Для этого сравниваемые выборки случайно разбиваем на 5 частей: четыре группы попеременно используется для построения модели, а по пятой проводится ее тестирование:

```
group.presence <- kfold(x = obs.Yes, k = 5)
group.background <- kfold(x = bg, k = 5)
spp <- 0
spa <- 0
sAUC <- 0
# Каждую группу выделяем в качестве тестовой
for (testing.group in 1:5) {
# Делим наблюдения на обучающую и тестовую последовательности
   presence.train <- obs.Yes[group.presence != testing.group, ]
   presence.test <- obs.Yes[group.presence == testing.group, ]
# Повторяем эту процедуру для pseudo-absence точек
   background.train <- bg[group.background != testing.group, ]
   background.test <- bg[group.background == testing.group, ]
# Построение модели распределения на обучающих данных
   me.model_cr <- maxent(x = ClimAltTri7.data, p = presence.train)
# Используем тестовые данные для оценки модели
   me.eval <- evaluate(p = presence.test, # Тестовые данные присутствия
                       a = background.test, # Тестовые данные отсутствия
                       model = me.model_cr, # Оцениваемая модель
           x = ClimAltTri7.data) # Растр переменных, используемых для модели
# Определение минимального порога для "presence"
   me.threshold <- threshold(x = me.eval, stat = "spec_sens")
   spp <- spp + sum(me.eval@presence > me.threshold)
   spa <- spa + sum(me.eval@absence < me.threshold)
   sAUC <- sAUC + me.eval@auc
}
```

```
c(spp, spp/nrow(obs.Yes)) # Правильное угадывание присутствия
[1] 83  0.8924731
c(spa, spa/1000) # Правильное угадывание отсутствия
[1] 787   0.787
sAUC/5
[1] 0.8808713
```

На основе рассчитанных показателей можно сделать два вывода: а) модель MaxEnt существенно более эффективна, чем BIOCLIM, и б) при тестировании модели на независимых данных в ходе кросс-проверки качество классификатора, оцененное по AUC, снижается весьма незначительно, что свидетельствует об устойчивом характере формируемых решений. Отмечается, что успешность работы алгоритма во многом зависит от выбора формы частных функций, объемов обеих выборок (*presence-only* и *background points*), предварительной фильтрации исходных данных, использования слоя коррекции и др. (Лисовский и др., 2020а). Но это – слишком тонкая материя для нашего ускоренного экскурса.

Использование случайных фоновых точек – это классическая процедура, которая известна как функция выбора ресурсов (*Resource Selection Functions* – Johnson, 1980), предполагающая сравнение текущих условий среды обитания с оценками доступности необходимых ресурсов для сообщества. Однако, поскольку часто очень трудно подтвердить факт отсутствие вида, было показано, что эта процедура оценивает не столько искомую вероятность присутствия вида, сколько неоднородность используемых эмпирических данных. В частности, показатели успеха предсказания отсутствия часто определяются "капризными ноликами", т.е. теми точками, где вид просто не может встречаться (Hastie, Fithian, 2013, Guisande et al., 2016). Поэтому, если доступны данные "присутствия-отсутствия" или, тем более, количественные оценки численности популяций, то целесообразно применять адекватные статистические методы.

4. Оценка индекса экологической пригодности методом виртуальных видов

Метод виртуальных видов связывают с понятием экологической ниши – «области в многомерном пространстве всех потенциальных переменных, так или иначе определяющих существование каждого вида и его численность» (G. Hutchinson, цит. по Пузаченко, 2004, с. 240). Селекция "потенциальных переменных" при наличии необходимого комплекта исходных данных для любого вида или их однородной группы – стандартная, хотя и не простая статистическая процедура. Модели распределения совокупности видов с одинаковым и известным откликом по отношению к факторам окружающей среды, в контексте SDM названных искусственными или "виртуальными" видами, могут быть полезным, а часто и единственным способом реализовать шаги, связанные с экологическим моделированием и статистическим анализом (Austin et al., 2006).

Если априори оценены ключевые параметры **функции ниши**, определяющие некий экологический оптимум, то распределение "виртуального вида" (Hirzel at al., 2001) полностью абстрагируется от данных экспедиционных исследований о встречаемости таксонов и моделирует ячеистую структуру в *n*-мерном пространстве, основываясь исключительно на факторах среды. Для каждой ячейки устанавливается количественно вероятность принадлежности к нише; что фактически является индексом пригодности среды обитания ($H \in [0,1]$ – *environmental suitability*) для всех видов, у которых экологический оптимум соответствует декларируемой функции ниши.

Пакет `virtualspecies` (Leroy at al., 2016) предусматривает представление функции ниши двумя способами:

1) определение функций отклика (таких как линейная, логистическая, квадратичная, гауссиана) для каждого из отобранных абиотических переменных, которые затем обобщаются в виде аддитивного или мультипликативного выражения (функция `generateSpFromFun`);

2) формирование модели анализа главных компонент (PCA) по всем переменным среды, определение положения отклика на каждой из двух главных латентных осей и оценка индекса экологической пригодности (функция `generateSpFromPCA`).

При определении функции ниши всегда надо учитывать проблему реалистичности экологических требований. Если анализируются несколько переменных среды, легко выбрать функции отклика, которые совместимы между собой. Например, не следует пытаться создать виртуальный вид, который требует средней температуры самого теплого месяца в 35°C, и температуры самого холодного месяца в -25°C, потому что такие условия вряд ли существуют на земле.

Определим функцию ниши на основе 7 биоклиматических показателей, взяв за основу эмпирическое распределение численности *Procladius ferrugineus.* Построим предварительно линейную модель для этого отклика с учетом предполагаемых парных эффектов взаимодействия некоторых предикторов.

```
var_clim <- c("bio1",  "bio3",  "bio8",  "bio15", "bio17", "alt", "tri")
df_m <- df.clim[,var_clim]
df_m$y_m <- log(df$ChPrc.f. + 1)
m <- lm(y_m ~ 0+(bio1*alt*tri)+bio3+bio8+bio15+bio17,data=df_m)
summary(m)
Coefficients:
               Estimate Std. Error t value Pr(>|t|)
bio1         -4.201e-02  9.036e-02  -0.465  0.64279
alt          -1.672e-01  5.454e-02  -3.066  0.00268 **
tri          -6.797e-02  6.117e-02  -1.111  0.26870
bio3         -1.478e-01  3.644e-01  -0.406  0.68574
bio8          5.115e-03  6.259e-03   0.817  0.41536
bio15         6.998e-02  1.078e-01   0.649  0.51759
bio17         8.496e-02  2.979e-02   2.852  0.00511 **
bio1:alt      3.743e-03  1.114e-03   3.361  0.00104 **
bio1:tri      1.047e-03  1.097e-03   0.954  0.34176
alt:tri       1.604e-03  6.369e-04   2.518  0.01309 *
bio1:alt:tri -3.314e-05  1.249e-05  -2.654  0.00902 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 2.919 on 121 degrees of freedom
Multiple R-squared:  0.7157,    Adjusted R-squared:  0.6898
F-statistic: 27.69 on 11 and 121 DF,  p-value: < 2.2e-16
```

Выполним перебор всех возможных субмоделей и найдем среди них оптимальную, доставляющую минимум информационного критерия Акаике. Для этого используем функции пакета `MuMIn` (см. сообщение «Селекция моделей…»)

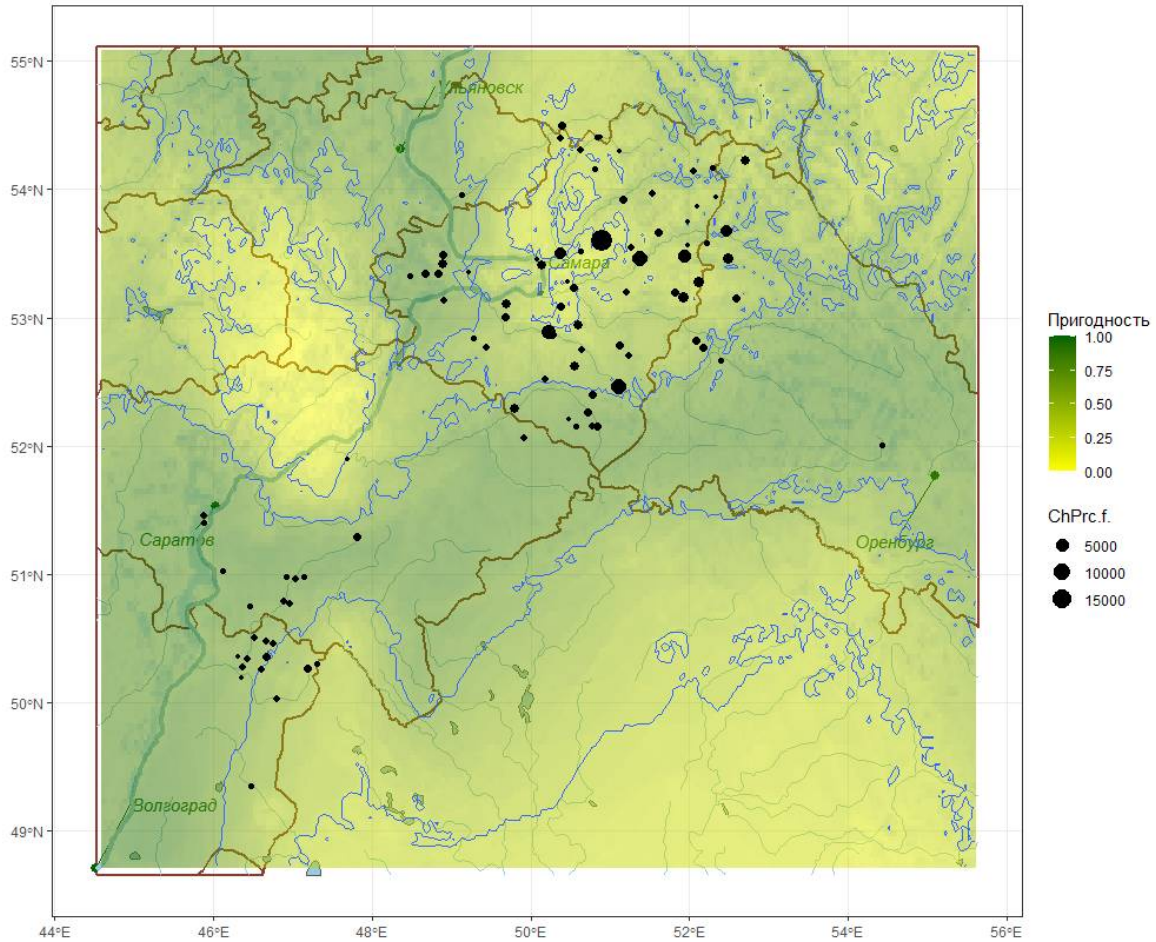
```
library(MuMIn)
options(na.action = "na.fail")
ms <- dredge(m)
subset(ms,delta < 1.8)
Model selection table
       alt     bi1     bi7      tri  alt:bi1  alt:tri   bi1:tri alt:bi1:tri df   logLik  AICc delta weight
73                  0.06550 -0.01547                                        3 -329.813 665.8  0.00  0.594
1996 -0.1573 -0.04345 0.07973 -0.04327 0.003534 0.001433 0.0005878  -2.991e-05  9 -323.549 666.6  0.76  0.406
Models ranked by AICc(x)
```

Для построения функции ниши используем выражение, полученное выше для оптимальной модели 1 (поскольку есть серьезные опасения, что полная модель и модель 1996 являются сильно переопределенными и потому неустойчивыми).

```
library(virtualspecies)
ClimAltTri2.data <- ClimAltTri7.data[[c("bio17", "tri")]]
responses <- formatFunctions(
   bio17 = c(fun = 'dnorm', mean = mean(dfR$bio17), sd = sd(dfR$bio17)),
   tri = c(fun = 'dnorm', mean = mean(dfR$tri), sd = sd(dfR$tri)))
# Генерация распределения виртуального вида
v.species <- generateSpFromFun(raster.stack = ClimAltTri4.data,
          parameters = responses, formula = 0.0655*bio17-0.01547*tri)
```

Вместо стандартного растрового изображения выведем карту с использованием средств `ggplot2()`:

```
library(tidyverse)
library(metR)
load(file="WB_map.RData")
dfVs <-  as.data.frame(cbind(coordinates(v.species$suitab.raster),
                             z=raster::values(v.species$suitab.raster)))
# Проведем отображение базовой карты и проведение изолиний
Basemap +
geom_tile(data = dfVs, aes(x = x, y = y, fill=z), alpha = 0.5) +
   scale_fill_gradient("Пригодность",low = "yellow", high = "darkgreen") +
        geom_contour(data = dfVs,aes(x = x, y = y, z=z), bins = 3) +
  geom_point(data = df_[df$ChPrc.f.>0,] ,aes(x = X , y = Y, size=ChPrc.f.))+
             theme_bw()
```



Области высокой экологической пригодности не вполне точно отражают ареал вида *Procladius ferrugineus*. Но карта была получена в соответствии с теми переменными, которые выбраны на основании построенной модели. Можно только предположить достаточную отвлеченность обоих предикторов от реальных условий существования гидробионтов и рассмотреть иные функции ниши.

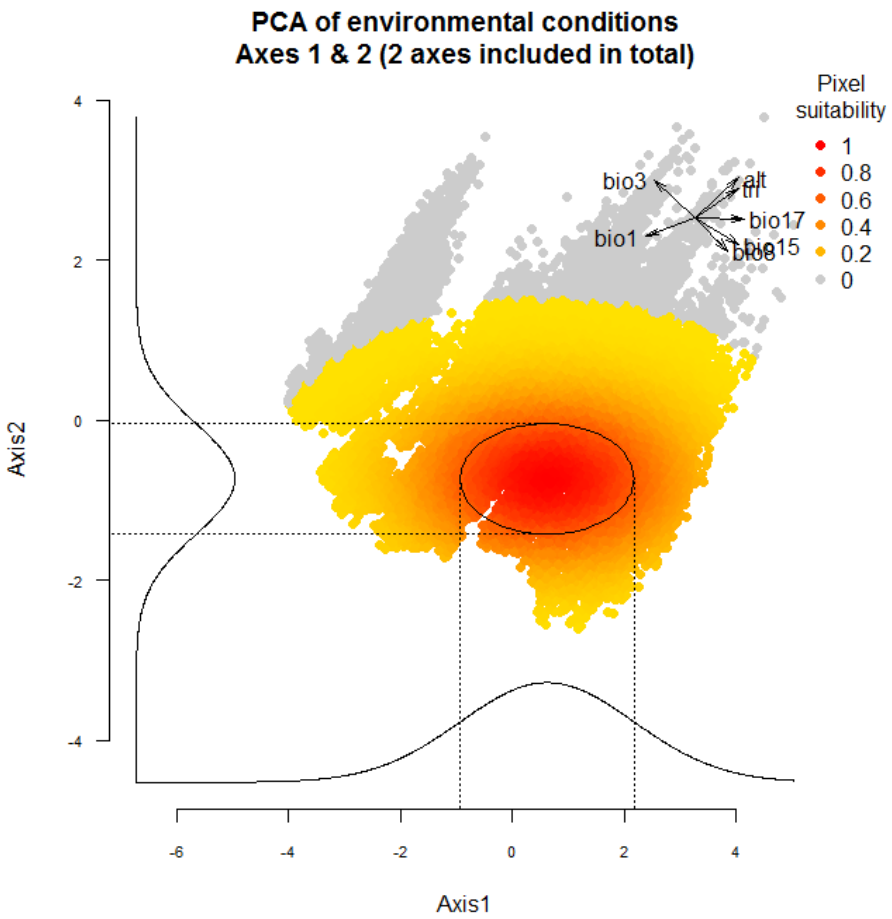
Если в анализе распределения виртуальных видов включено больше 10 переменных, то задача определения функции ниши намного сложнее: создавая вид, который зависит от 5 различных переменных температуры, 3 переменных осадков и 2 переменных свойств почвы, почти невозможно предугадать, будут ли реалистичны предполагаемые функции отклика относительно условий окружающей среды. Для этого реализован второй подход, который заключается в анализе главных компонент (PCA) всех переменных среды из исходного растра, а затем оценивается статистическое распределение вида по двум главным осям.

```
pca.species <- generateSpFromPCA(raster.stack = ClimAltTri7.data,
                                 plot=FALSE)
Virtual species generated from 7 variables:
 bio1, bio3, bio8, bio15, bio17, alt, tri
- Approach used: Response to axes of a PCA
- Axes:  1, 2 ;  74.29 % explained by these axes
- Responses to axes:
   .Axis 1  [min=-4.01; max=5.03] : dnorm (mean=-3.05013; sd=2.760278)
   .Axis 2  [min=-2.61; max=3.78] : dnorm (mean=0.1793298; sd=1.163532)
- Environmental suitability was rescaled between 0 and 1
# Извлекаем нагрузки на оси из объекта pca.species
score <- as.matrix(pca.species$details$pca$c1)
t(score)
         bio1       bio3       bio8      bio15       bio17       alt       tri
CS1 -0.4327258 -0.3637228  0.2769861  0.3794261  0.42469990 0.3767879 0.3707149
CS2 -0.2456764  0.4803054 -0.4328006 -0.3507997 -0.01996673 0.5104800 0.3709238
```

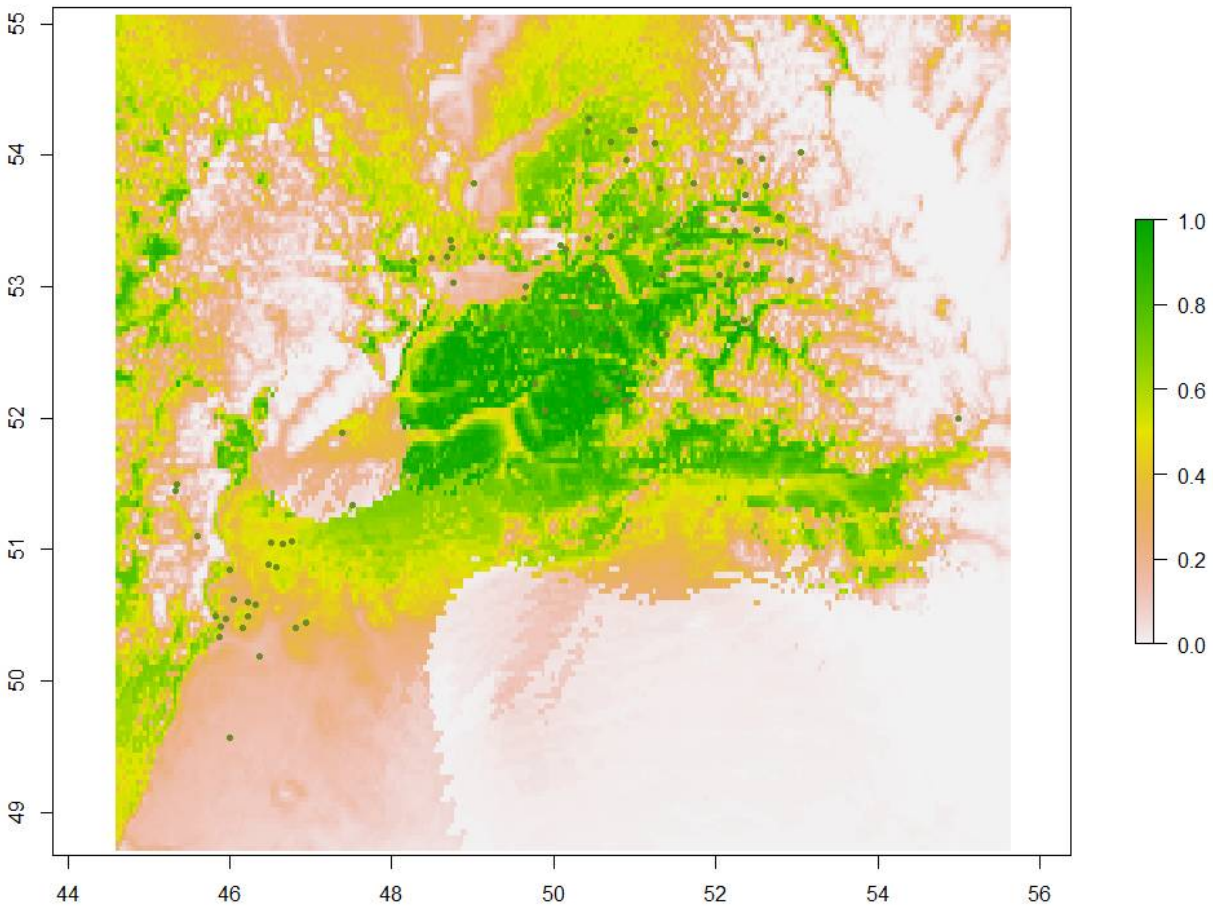
При преобразовании исходного пространства переменных к координатам по двум главным осям редукции `CS1-CS2` объясняется 74.5% совокупной вариации данных. Положительные значения 1-й главной оси определяются объемом осадков (`bio17`), а отрицательные – уровнем среднегодовой температуры (`bio1`). Положительные значения 2-й оси определяются высотой и вертикальной расчлененностью рельефа (`alt, tri`).

Объем и положение экологической ниши определяется распределением эмпирических значений факторов среды (с учетом их нагрузок) по каждой оси латентных координат. По умолчанию – это распределение переменных по географической сетке всего региона, т.е. по 41385 ячейкам исходного растра `ClimAltTri7.data`. Но в нашем случае интерес представляет фундаментальная ниша, характерная для вида *Procladius ferrugineus*. Найдем параметры распределения проекций факторов среды на латентные оси `CS1-CS2` в точках, где вид встретился.

```
# Выполняем стандартизацию данных
ma <- apply(dfR[,-(1:2)], 2, mean)
sda <- apply(dfR[,-(1:2)], 2, sd)
df_ms <- df.clim[df$ChPrc.f. !=0,var_clim]
df_ms <- as.matrix(df_ms)
for (i in 1:7) df_ms[,i] <- (df_ms[,i]-ma[i])/sda[i]
# Находим среднее и стандартные отклонения для точек встречаемости вида
pcsnew <- df_ms %*% score
apply(pcsnew, 2, mean)
       CS1        CS2
 0.7071814 -0.8047507
apply(pcsnew, 2, sd)
      CS1       CS2
1.4293504 0.6546431
pca.newsp <- generateSpFromPCA(raster.stack = ClimAltTri7.data, plot=FALSE,
        means = c(0.7071814, -0.8047507), sds = c(1.4293504, 0.6546431))
plotResponse(pca.newsp)
```

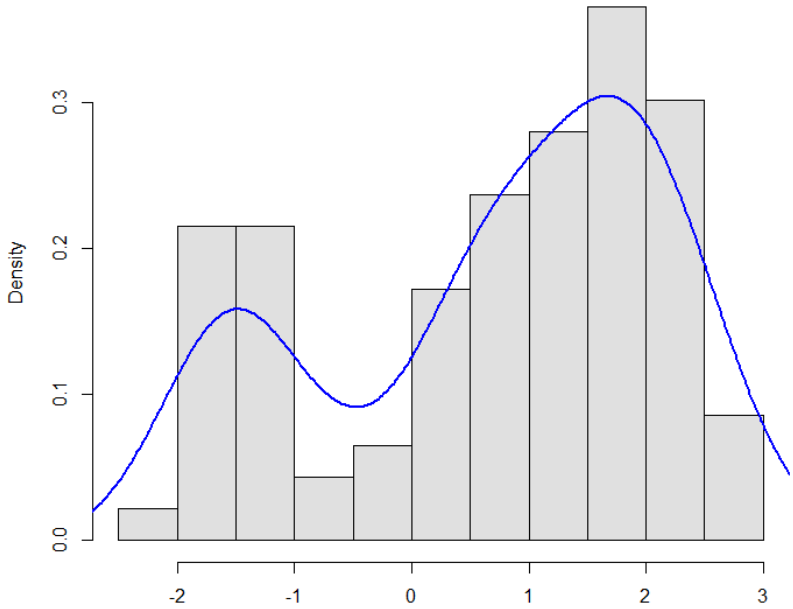



```
plot(pca.newsp)
points(obs.Yes$X, obs.Yes$Y, col = "olivedrab", pch = 20, cex = 0.95)
```



На приведенной карте экологической пригодности географическая ниша для размещения вида *Procladius ferrugineus* окрашена в зеленый цвет. Правда, мы предположили, что проекции факторов на главные оси распределены по нормальному закону. Это оказалось не вполне верным: на построенной кривой ядерной плотности отчетливо видны два горба…

```
# Гистограмма и ядерная плотность координат по 1-й главной оси
hist(pcsnew[,1], freq = FALSE, breaks=8,col="grey88")
lines(density(pcsnew[,1]), lwd = 2, col="blue")
```



Действительно, точки на карте, где обнаружен вид *Procladius ferrugineus,* составляют два достаточно обособленных кластера на северо-востоке и юго-западе с различающимися условиями среды по геоклиматическим показателям. Алгоритм `generateSpFromPCA` этого не учитывает и область максимальной экологической пригодности вида разместилась аккуратно между этими двумя кластерами.

Настало время объясниться, почему мы до сих пор при построении моделей использовали исключительно пространственно распределенные геоклиматические показатели и игнорировали локальные характеристики биотопов (состав химических ингредиентов и гидрологические параметры водотока, тип донного грунта и т.д.). Во-первых, все функции пакетов, использованные в разделах 2-4, работают только с растровой информацией, представленной в ячейках ("пикселах") равномерной сетки достаточно высокого разрешения, а представить на такой сетке данные о совокупности биотопов речной сети нам показалось невозможным. Во-вторых, пространственное распределение отдельных видов, как правило, автокоррелировано: появление таксона в некоторой точке увеличивает вероятность его обнаружения в соседних экотопах.

Разумеется, любая река по-своему уникальна, и поэтому условия существования локальных сообществ и видовой состав гидробионтов в каждой из них может быть совершенно различен. И на уровне отдельной малой или средней реки (в линейном масштабе от 10 до 300 км) основные задачи связаны с моделированием распределения таксономического структуры гидробионтов по продольному профилю водотока, что обусловлено непрерывно-прерывистым (пунктирным) градиентом изменения важнейших гидрологических условий и качества водной среды.

Однако и на региональном уровне можно обнаружить специфические пространственные закономерности, характерные для большинства мета-сообществ в рамках крупномасштабной экосистемы, происходящие под влиянием ландшафтных,

климатических или геоморфологических факторов, что определяет необходимость комплексного характера проведения биосферных исследований. Действительно, планктонные и, отчасти, бентосные организмы способны рассеиваться на различных стадиях метаморфоза, либо перемещаться по градиенту течения в пределах речной сети на сотни километров. Отмечено, что потоки миграции инвазивных видов могут быть, в известной степени, стационарными, и таксономическая структура локальных ценозов в зоне расселения чужеродных организмов может приобрести дополнительное сходство. В рамках крупного региона часто удается выделить относительно однородные области с одинаковым составом ландшафтных элементов или уровнем антропогенного воздействия (сельскохозяйственной нагрузки), что приводит к сходству видовой структуры сообществ (Manni et al., 2004). Наконец, для каждой таксономической группы объективно существует некоторый географический вектор, относительно которого встречаемость отдельных видов может статистически значимо изменяться (см., например, теорию *широтного градиента разнообразия* – Koleff, Gaston, 2001).

Поскольку пространственный градиент земной поверхности (и выраженность автокорреляции) определяется в основном тремя факторами: температурой, осадками и высотой, эти обстоятельства и обусловили наш интерес к построенным выше моделям. Но в дальнейших разделах мы обратимся к расширенному списку предикторов.

5. Модели, основанные на данных о популяционной плотности видов

Как упоминалось во введении, большой коллектив авторов (Norberg, 2019) выполнил многоплановый анализ 33 моделей SDM, построенный на данных типа наличие-отсутствие. Однако, в практике гидробиологических исследований после тщательного разбора проб аналитику доступны данные о популяционной плотности каждого вида (для бентоса – удельная численность и биомасса организмов на кв. м дна водотока). Если выполнена повторность проб и вид не встретился ни в одной из 10 из них (как в среднем в нашем случае), то предположение об отсутствии вида выглядит вполне реалистичным.

Наличие количественных данных об обилии видов требует привлечения соответствующих статистических методов, выполняющих построение моделей в условиях непрерывной шкалы отклика. Мы не планируем довести число использованных моделей до 33 и скрупулезно продиагностировать полученные результаты, а просто покажем примеры использования нескольких основных методов. Все эти модели обсуждались ранее (Шитиков, Мастицкий, 2017, глава 4), что избавляет нас от необходимости давать подробные комментарии.

Будем оценивать пространственное распределение целой группы личинок комаров-звонцов – Orthocladiinae, которые слывут прекрасными индикаторами загрязнения (без учета вида *Cricotopus* gr. *Sylvestris,* считающегося эврибионтным). В качестве факторов среды будем использовать как 7 геоклиматических показателей, фигурировавших в предыдущих разделах, так и 4 гидрохимических показателя из таблицы `df_var`.

```
var_clim <- c("bio1",  "bio3",  "bio8",  "bio15", "bio17", "alt", "tri")
df_m <- df.clim[,var_clim]
df_m <- cbind(df_m,df_var[,-1])
y_m <- log((df$ORTHOCLADIINAE - df$ChCri.s.)/df$Проб +1)
```

Построим вначале обычную линейную модель.

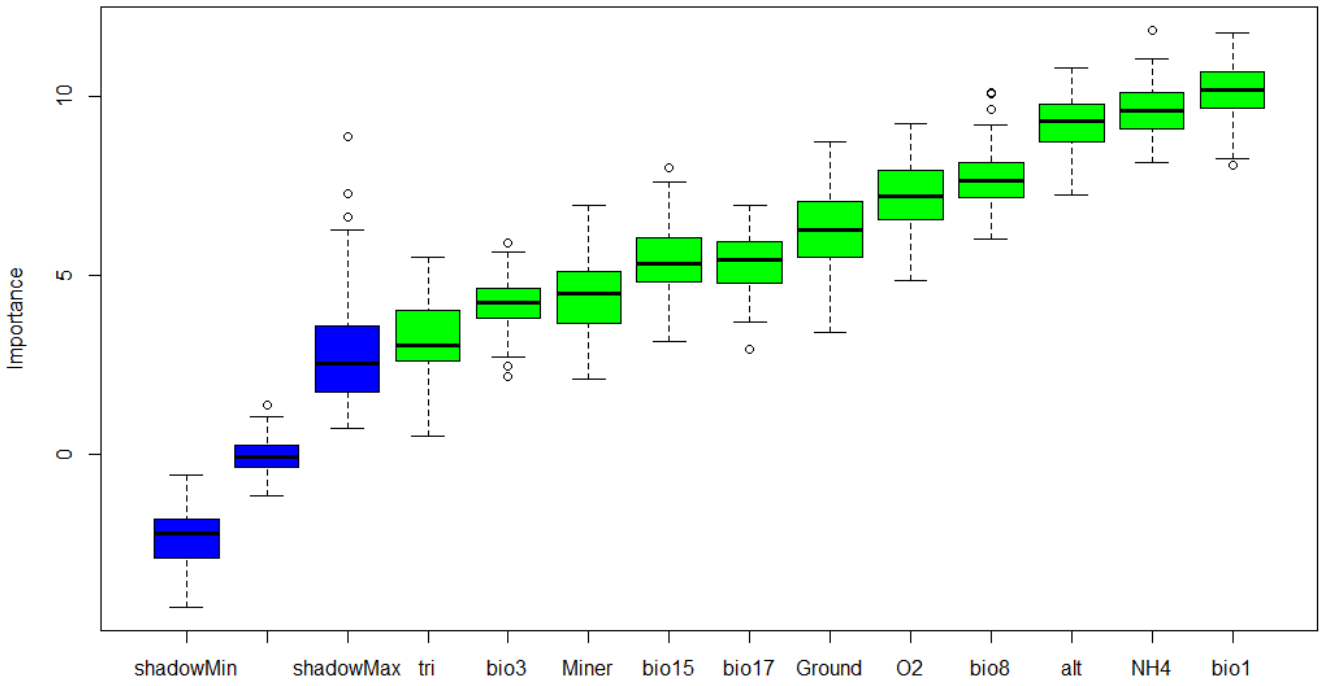
```
m <- lm(y_m ~ .,data=df_m)
summary(m)
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.018e+00  1.217e+01   0.577 0.565332
bio1         3.293e-02  5.264e-02   0.626 0.532757
bio3        -4.618e-01  3.462e-01  -1.334 0.184711
bio8         7.443e-03  4.312e-03   1.726 0.086927 .
bio15        1.059e-01  9.206e-02   1.150 0.252242
bio17       -2.341e-02  4.067e-02  -0.576 0.565907
alt          1.342e-02  6.065e-03   2.213 0.028792 *
tri          7.099e-03  4.409e-03   1.610 0.109987
Ground      -9.682e-02  1.196e-01  -0.809 0.419860
Miner        7.166e-06  3.022e-05   0.237 0.812948
NH4          2.062e-01  5.613e-02   3.673 0.000359 ***
O2           1.316e-02  6.724e-03   1.957 0.052643 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.753 on 120 degrees of freedom
Multiple R-squared:  0.3404,    Adjusted R-squared:   0.28
F-statistic: 5.631 on 11 and 120 DF,  p-value: 3.049e-07
```

В этой модели только 4 предиктора из 11 оказались значимыми. Но являются ли остальные 7 переменных бесполезными для выполнения прогноза? Используем алгоритм "Борута" (*Boruta* – Kursa, Rudnicki, 2010), который оценивает меру информативности каждого фактора в виде частной ошибки прогноза, вызванной исключением этой переменной из модели. Метод выполняет проверку H_0 путем рандомизации с использованием 99 итераций построения моделей "случайного леса", состоящих из 500 иерархических деревьев. Важность (*importance*) каждого предиктора оценивается по Z-критерию, отражающему снижение ошибки прогнозирования отклика при замене эмпирического вектора переменной на случайный вектор.

```
library(Boruta)
Ort.Boruta <- Boruta(x=df_m, y=y_m, doTrace = 2, ntree = 500)
plot(Ort.Boruta)  , xlab = "", xaxt = "n")
```



Результаты показывают, что все переменные являются полезными при моделировании и снижают ошибку прогноза. Наиболее важными факторами среды являются среднегодовая температура, содержание ионов аммония и высота.

Тестирование моделей будем проводить единообразно с использованием функций пакета `caret`. Оценивать каждую модель будем с использованием кросс-проверки с разбиением исходной выборки на 5 частей. Прогнозы прологарифмированной численности ортокладеин, полученные каждым методом, будем накапливать в таблице `df_Pred`.

```
df_Pred <- as.data.frame(y_m)
library(caret)
ctrl <- trainControl(method = "cv",number = 5)
```

1 модель. Начнем с тестирования обычной *линейной* модели:

```
 (lm.cv <- train(x=df_m, y=y_m, method = 'lm',trControl = ctrl))
132 samples
 11 predictor
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 106, 105, 106, 106, 105
Resampling results:
  RMSE      Rsquared   MAE
  1.946237  0.2133734  1.476744
df_Pred$lm <- predict(lm.cv)
```

Обратим внимание, что полученные в ходе кросс-проверки корень из среднеквадратичной ошибки (`RMSE`) и квадрат коэффициента детерминации (`Rsquared`) несколько хуже, чем полученные по полной эмпирической выборке.

2 модель – регрессия *lasso*. Чтобы построить корректную финальную модель, нужно предварительно оценить значение параметра регуляризации `lambda`:

```
grid.train = seq(0.02,0.3,length=8)
(lasso.cv <- train(x=df_m, y=y_m, method='glmnet', trControl = ctrl,
    tuneGrid = expand.grid(.lambda = grid.train, .alpha = 1)))
  lambda  RMSE      Rsquared   MAE
  0.02    1.971672  0.2026801  1.542688
  0.06    1.922714  0.2016568  1.522147
  0.10    1.892988  0.2093997  1.506613
  0.14    1.878409  0.2173164  1.497068
  0.18    1.879796  0.2216914  1.492756
  0.22    1.895171  0.2195878  1.495927
  0.26    1.919752  0.2078134  1.509344
  0.30    1.950388  0.1762972  1.531383

Tuning parameter 'alpha' was held constant at a value of 1
RMSE was used to select the optimal model using the smallest value.
The final values used for the model were alpha = 1 and lambda = 0.14.

df_Pred$lasso <- predict(lasso.cv)
```

Регуляризация не внесла существенных улучшений в качество линейной модели.

3 модель. Метод *частных наименьших квадратов* PLS (Partial Least Squares, или Projection into Latent Structure) использует разложение исходных предикторов по осям главных компонент и необходимо оценить, сколько латентных осей следует использовать:

```
(pls.cv <- train(x=df_m, y=y_m, method = "pls",
        tuneLength = 8, trControl = ctrl,  preProc = c("center", "scale")))
  ncomp  RMSE      Rsquared   MAE
  1      1.854994  0.2737906  1.444249
  2      1.830916  0.2911452  1.472709
  3      1.893221  0.2405738  1.513215
  4      1.941627  0.2326130  1.546910
  5      1.960264  0.2330585  1.545551
  6      1.961641  0.2458935  1.533536
  7      1.963962  0.2445877  1.535859
  8      1.963982  0.2450431  1.536762

RMSE was used to select the optimal model using the smallest value.
The final value used for the model was ncomp = 2.

df_Pred$pls <- predict(pls.cv)
```

Оказалось, что для построения модели PLS достаточно двух первых главных компонент.

4 модель. Алгоритм случайного леса (*Random Forest*) на каждой итерации построения дерева случайным образом выбирает *m* предикторов из *p* подлежащих рассмотрению, для чего предварительно оценивается параметр `mtry`:

```
(rf.cv <- train(x=df_m, y=y_m, method = 'rf',
     preProc = c('center', 'scale'),
     tuneGrid = expand.grid(.mtry = 2:5)))
  mtry  RMSE      Rsquared   MAE
  2     1.818787  0.2614807  1.428363
  3     1.835089  0.2491752  1.437449
  4     1.841856  0.2438468  1.440213
  5     1.850613  0.2388123  1.445026

RMSE was used to select the optimal model using the smallest value.
The final value used for the model was mtry = 2.

df_Pred$rf <- predict(rf.cv)
```

5 модель. Метод *опорных векторов с радиальным ядром* SVMR нуждается в предварительной оценке двух параметров – `C` и `sigma` (допустимый штраф за нарушение границы зазора и параметр радиальной функции).

```
(svR.cv <- train(x=df_m, y=y_m,    method="svmRadial", trControl = ctrl,
    tuneGrid = expand.grid(sigma = seq(0.005,0.035,length=4), C = 1:3)))
  sigma  C  RMSE      Rsquared   MAE
  0.005  1  1.891323  0.1744644  1.510216
  0.005  2  1.872223  0.1827188  1.496165
  0.005  3  1.871942  0.1779798  1.498454
  0.015  1  1.839270  0.2145067  1.485760
  0.015  2  1.829413  0.2251399  1.480293
  0.015  3  1.834823  0.2290493  1.483771
  0.025  1  1.827377  0.2224004  1.484008
  0.025  2  1.826175  0.2316667  1.474194
  0.025  3  1.838330  0.2282433  1.478010
  0.035  1  1.826784  0.2219722  1.483039
  0.035  2  1.853281  0.2114483  1.493938
  0.035  3  1.876900  0.2040538  1.515748

RMSE was used to select the optimal model using the smallest value.
The final values used for the model were sigma = 0.025 and C = 2.
```

```
df_Pred$svR <- predict(svR.cv)
```

Наконец, предпримем попытку осуществить *коллективный* прогноз.

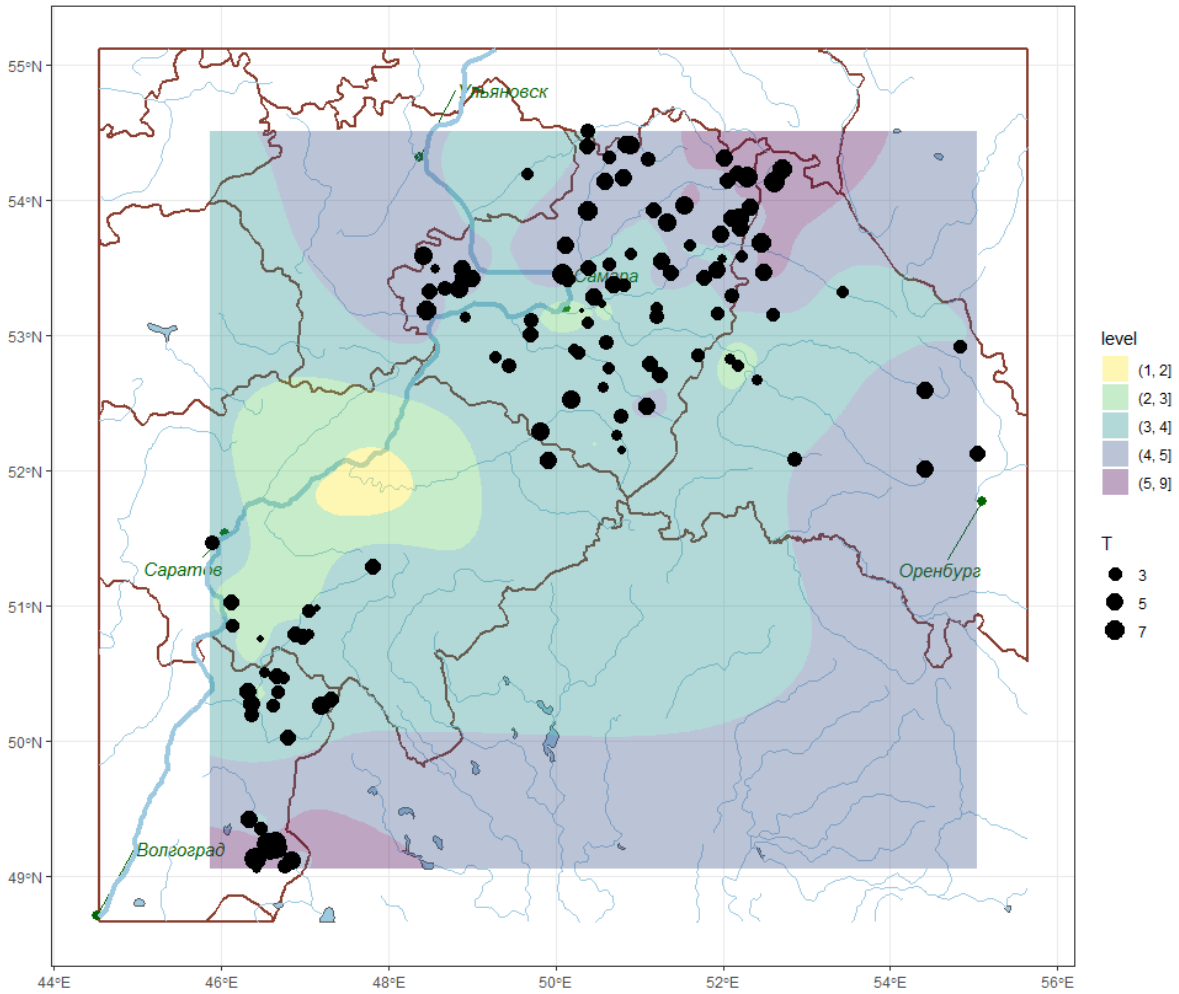
```
# Значения среднеквадратичной ошибки для каждой модели
w <- c(min(lm.cv$results$RMSE), min(lasso.cv$results$RMSE),
       min(pls.cv$results$RMSE), min(rf.cv$results$RMSE),
       min(svR.cv$results$RMSE))
[1] 1.946237 1.972940 1.830916 1.818787 1.826175
# Значения квадрата коэффициента детерминации для каждой модели
r <- c(max(lm.cv$results$Rsquared), max(lasso.cv$results$Rsquared),
       max(pls.cv$results$Rsquared), max(rf.cv$results$Rsquared),
       max(svR.cv$results$Rsquared))
[1] 0.2133734 0.2216914 0.2911452 0.2614807 0.2316667
```

Все модели дали не слишком отличающиеся друг от друга результаты прогноза, поэтому построение коллектива не имеет особого смысла. Тем не менее, обозначим веса и проведем оценку средневзвешенного прогноза:

```
weight <- r/sum(r)
[1] 0.1749884 0.1818100 0.2387694 0.2144414 0.1899908
CombyPred <- apply(df_Pred[,-1],1, function (x) sum(x*weight))
```

Отобразим для сравнения результаты прогноза и эмпирические данные на карте. При этом у нас имеется тривиальная проблема: мы имеем прогноз в 132 локальных точках на карте и необходимо осуществить интерполяцию модельных значений на всю территорию. Выполним это методом иерархических базисных сплайнов, доступном в пакете MBA:

```
toMap <- df[,4:5]
toMap$Z <- CombyPred
toMap$T <- df_Pred[,1]
library(MBA)
library(reshape2)
toMap <- as.data.frame(toMap)
mba.int <- mba.surf(toMap[,-4], 300, 300, extend=TRUE)$xyz.est
dimnames(mba.int$z) <- list(mba.int$x, mba.int$y)
mba.output <- melt(mba.int$z, varnames = c('X', 'Y'), value.name = 'Z')
library(tidyverse)
library(metR)
load(file="WB_map.RData")
brk <- c(0, 1, 2, 3, 4, 5, 9)
Basemap +
   geom_contour_filled(data = mba.output,aes(x = X, y = Y, z=Z),
                  breaks = brk) +
   scale_fill_viridis_d(begin = 1, end = 0, alpha = 0.35) +
   geom_point(data = toMap[toMap$T>0,],aes(x = X , y = Y, size=T)) +
        theme_bw()
```



Нельзя сказать, что построенная карта в полной мере удовлетворит эколога, но, принимая во внимание отсутствие явных закономерностей локализации популяций ортокладеин, мы получили примерно то, что и должны были получить.

6. Пакет HMSC – комплексное иерархическое моделирование сообществ видов

Встречаемость (и/или обилие) вида в общем случае определяется целой совокупностью экзогенных и эндогенных процессов в экологических сообществах:

1. Центры статистических распределений совокупности независимых пространственно-непрерывных (геофизических) или локальных (биотопических) факторов задают размерность и структуру гиперпространства фундаментальной ниши для каждого из видов. Чем больше расстояние произвольного местообитания от этого оптимума, тем ниже экологическая пригодность среды и меньше вероятность встретить вид.
2. Географическое пространство обладает свойством автокоррелированности, т.е. значения обилия вида в точках, расположенных близко друг к другу, вероятнее всего, будут более сходными, чем у выборочных единиц, расположенных далеко друг от друга.
3. Живые организмы каждого вида обладают определенными характеристиками (*traits*), которые позволяют им приспосабливаться к конкретным условиям окружающей среды (особенности питания, измененные морфологические признаки,

масса тела и т.д.). Зная эти характеристики, можно оценить вероятность встречаемости вида в анализируемом биотопе.

4. Между видами в сообществе существуют взаимодействия, в результате которых численности популяций могут снижаться (конкуренция) или увеличиваться (симбиоз). Здесь важная проблема состоит в том, что выводы о совместном сосуществовании, определяемом межвидовыми взаимодействиями, смешиваются с эффектами, порожденными совместной вариацией отклика видов на абиотические изменения.
5. На ассоциативность видов могут влиять филогенетические отношения: близкие в таксономическом отношении популяции статистически чаще вступают в сложные экологические связи между собой, чем далекие.

Все перечисленные факторы действуют не в отдельности каждый сам по себе, а в составе взаимосвязанного комплекса, в результате чего анализируемый отклик (т.е. встречаемость вида в каждой точке) определяется сложным характером их совокупного влияния.

Модели SDM были в основном разработаны для моделирования ареала только одного вида, в то время как часто возникает задача оценить совместное распределение многих видов, образующих сообщества (Clark et al., 2014;Warton et al., 2015). Один из возможных подходов – сложение моделей распределения (*stacked* SDM, SSDM), где на первом этапе строится совокупность моделей для отдельных видов, а затем их результаты комбинируются (Calabrese et al., 2014). В отличие от него, другой обобщенный способ анализа (*joint* SDM, JSDM) объединяет видовой уровень данных модели в одну модель, которая одновременно подстраивается под структуру всего сообщества. Это позволяет не только выявлять межвидовые ассоциации, но и соотнести полученные закономерности с характеристиками видов (Abrego et al., 2017), их филогеническими особенностями или паттернами совместного сосуществования (Pollock et al., 2014). Наконец, класс моделей SDFA (*spatial dynamic factor analysis* – Thorson et al., 2015) рассматривает распределение структуры сообществ под влиянием факторов среды не только в пространстве, но и во времени.

Одна из попыток учесть все вышеперечисленные факторы при построении JSDM воплощена идеологически в составе платформы HMSC (*иерархическое моделирование сообществ видов* или *Hierarchical Modelling of Species Communities*), разработанной О. Оваскайненом с соавторами. В ее математической основе положена одна из версий моделей GLMM, которая по статистической терминологии трактуется как многомерная иерархическая обобщенная линейная модель со смешанными параметрами, основанная на байесовской процедуре их оценки. Методический материал полно представлен в книге (Ovaskainen, Abrego, 2020) и предыдущих статьях этого авторского коллектива (Ovaskainen et al., 2016а, 2016б, 2017; Tikhonov et al., 2017, 2020), а все необходимые вычисления можно получить с использованием R-пакета `HMSC`.

6.1. Описание статистической модели HMSC

Типичный набор данных, полученный в ходе экологических исследований сообществ, включает совокупность видов $j = 1...n_s$, выявленных на множестве n_y биотопов (строже говоря – в точках отбора проб, *sampling units*), $i = 1... n_y$. Используемая обобщенная линейная смешанная модель GLMM может быть применена к различным показателям обилия видов y_{ij} (наличие/отсутствие, количество, биомасса, покрытие и т. д.) путем включения различных функций связи и постулирования законов распределения ошибок. В контексте HMSC выборочные данные подгоняются многомерной моделью, т.е. число переменных отклика совпадает с числом видов n_s. Для каждого вида задается статистическое распределение $y_{ij} \sim \mathrm{D}\{L_{ij}, \sigma^2_j)$, где L_{ij} – математическое ожидание плотности вида j в точке i, а σ^2_j – параметр дисперсии (не используется в случае распределения Пуассона или Бернулли). В случае нормального

распределения значение L_{ij} моделируется как линейная функция от двух групп предикторов, представляющие фиксированные и случайные факторы:

$$L_{ij} = \sum_{k=1}^{n_c} x_{ij}\beta_{jk} + \varepsilon_{ij}\,, \quad \text{где} \quad \varepsilon_{ij} = \sum_{h=1}^{n_f} \eta_{ih}\lambda_{jh}(z_{i.})\,. \qquad (1)$$

Первый член выражения (1), моделирующий влияние фиксированных факторов, является обычной линейной регрессией, где x_{ik} – значение *к*-й переменной окружающей среды, наблюдаемое в точке i, $k = 1...n_c$, а β_{jk} – коэффициент регрессии, представляющий долю линейного отклика вида j на эту ковариату. Чтобы обеспечить параметризацию модели с разреженными данными или редкими видами, принимается распределение коэффициентов регрессии как $\beta_{j.} \sim N(\mu, \mathbf{V})$, где вектор μ является оценкой средней реакции вида на измеренные ковариаты, а дисперсионно-ковариационная матрица **V** соответствует вариации отдельных видов относительно математического ожидания. Здесь и далее точка в выражении $\beta_{j.}$ означает, что индекс k пробегает все значения от 1 до n_c для каждого фиксированного j.

Вклад совокупности случайных факторов, включая пространственную автокорреляцию и межвидовые взаимодействия, моделируются вторым членом ε_{ij}, который представляет собой сумму произведений n_f латентных факторов и нагрузок, $h = 1...n_f$. Здесь η_{ih} – это значение фактора для выборочной точки i, а $\lambda_{jh}(z_{i\cdot})$ – факторная нагрузка на вид j со стороны латентного фактора h, обобщающего произвольный набор предикторов $z_{i.}$. Если, в частности, принять, что $\lambda_{jh}(z_{i.}) = \sum_{k=1}^{n_c} x_{ik}\lambda_{jhk}$, то структура ковариаций между видами ε_{ij} становится функцией состояния окружающей среды, определяемого исходным набором переменных-ковариат x. Некоторые случайные факторы могут быть связаны с вложенной структурой плана исследований (например, бассейн водохранилища -> река -> точка отбора проб), поэтому рассматриваемая модель трактуется как иерархическая.

Коэффициенты модели (1) рассчитываются по данным наблюдений x с использованием байесовской методологии, которая основана на итеративном процессе подстройки исходных (априорных) оценок модельных параметров $\boldsymbol{\theta}$ и получении их результирующего (апостериорного) распределения. Этот процесс реализуется методом построения длинных итеративных последовательностей нескольких марковских цепей Монте-Карло (MCMC), для которых распределение переходов определяется функцией $P(\boldsymbol{\theta}|\mathbf{Y},\mathbf{X})$. Процесс моделирования часто довольно длительный и продолжается до тех пор, пока распределение текущих значений процесса не приблизится к некоторому стационарному распределению. Для проверки сходимости цепей используются приемы визуальной и формальной диагностики. Для проверки адекватности модели и сравнения ее различных вариантов используется алгоритм перекрестной проверки.

6.2. Связь модели с основными теоретическими конструкциями экологии сообществ

После построения и диагностики параметризованная модель HMSC (как и любая JSDM) может использоваться для объяснения экологических процессов в сообществах и/или для прогнозирования. Связи информационной структуры платформы HMSC с основными задачами экологии сообществ представлены на рис. 1. Прямоугольники включают обозначения матриц исходных данных, а эллипсы – вычисляемые параметры модели (1), которые могут быть использованы для анализа структуры экологических ниш и межвидовых взаимодействий в сообществе.

Часть коэффициентов $\beta_{j.} \sim N(\mu, \mathbf{V})$ модели HMSC, описывающих фиксированные эффекты, определяют, в какой мере изменчивость факторов **X** окружающей среды влияет на встречаемость и/или обилие видов. Каждый вид имеет свой вектор β-параметров, ограничивающий некоторый объем гиперпространства, а значит, и свою экологическую нишу. Однако границы ниши определяются не только параметрами внешнего воздействия, но и изменчивостью внутрипопуляционных характеристик **Г**

(*species-specific traits*), таких как размер тела, морфологические особенности или тип питания у животных, размер семян или жизненная форма у растений и т.д.

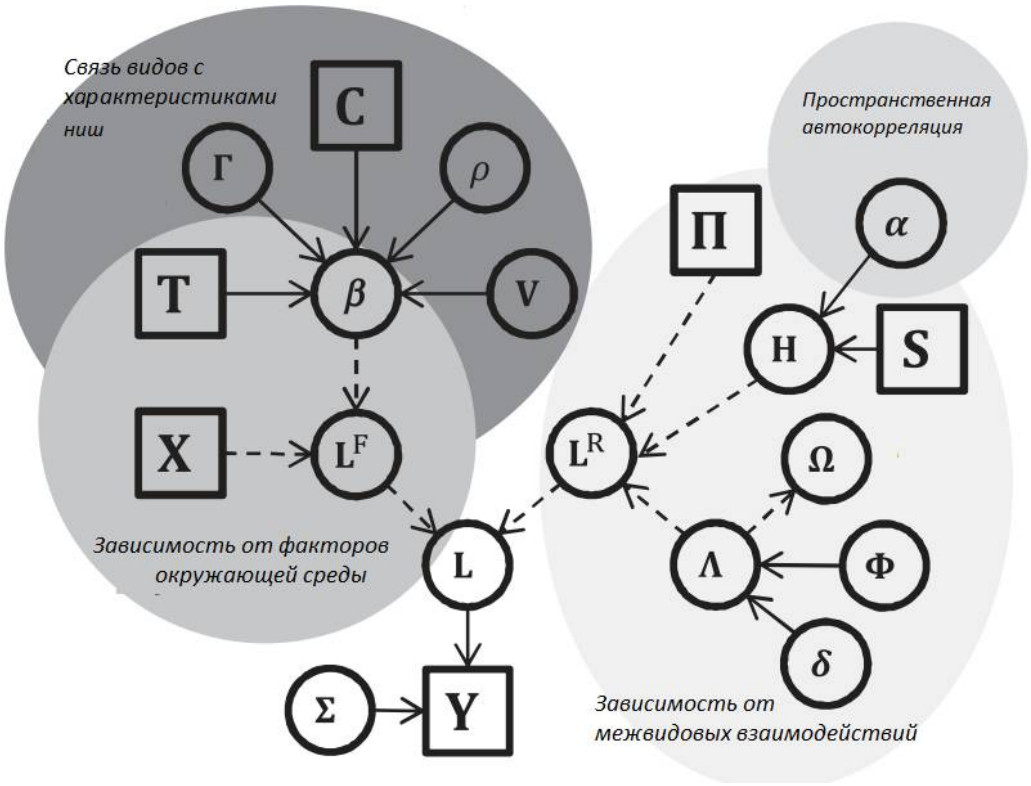


Рис. 1. Связи между теоретическими конструкциями экологии сообществ и статистической структурой платформы HMSC (Ovaskainen, Abrego, 2020).
Матрицы исходных данных: **Y** – обилие видов, **X** – факторы среды, **T** – свойства видов, **C** – филогенетические корреляции, **П** – план исследования, **S** – географические координаты; Переменные и параметры модели: **L** – линейные предикторы, $\mathbf{L}^F$ – фиксированные эффекты, $\mathbf{L}^R$ – случайные эффекты, β – ниши видов, **Г** – влияние характеристик видов в нише, ρ – филогенетический сигнал в нише, **V** – остаточная ковариация видов в нише, **H** – факторные нагрузки биотопов, α – пространственная шкала биотопов, **Λ** – факторные нагрузки видов, **Ω** – матрица объединения видов, **Φ** – локальные потери нагрузок видов, δ – глобальные потери нагрузок видов, **Σ** – матрица остаточной дисперсии.

Другим важным фиксированным эффектом, определяющим разбиение сообщества на экологические ниши, является филогенетическое родство между видами. Для того, чтобы структурировать ниши по этому признаку, филогенетическое дерево преобразуется в корреляционную матрицу **C** $n_s \times n_s$, элементы которой (c_{ij} = 0÷1) оценивают филогенетические корреляции, определяемые как долю общего эволюционного времени для каждой пары видов *i* и *j*. HMSC реализует филогенетическую корреляционную модель как $\beta_f \sim N(\mu_f, \mathbf{W})$, где $\mathbf{W} = \rho\,\mathbf{C} + (1 - \rho)\mathbf{I}$, а вычисляемый параметр ρ = 0÷1 измеряет силу филогенетического сигнала. Если предположить, что ниши полностью филогенетически структурированы, то ρ = 1 и коэффициенты модели имеют многомерное нормальное распределение $\beta_f \sim N(\mu_f, \mathbf{C})$. Эта модель имеет одинаковое ожидание μ_f для всех видов, но предсказывает, что филогенетически близкие виды в среднем будут иметь меньший статистический разброс, чем филогенетически отдаленные виды.

Совокупность случайных эффектов HMSC (показана справа на рис. 1) моделирует влияние различных биотических или абиотических факторов на вариацию отклика **Y** (не изменяя его математического ожидания). В большинстве случаев при реализации

плана исследований выборочные точки связаны с пространственными координатами, и тогда зависимость между остатками ε_{ij} обусловлена явлением, называемым пространственной автокорреляцией (наблюдения в точках, расположенных близко друг к другу, вероятнее всего, будут более сходными, чем для выборочных единиц, расположенных далеко друг от друга). HMSC моделирует любую автоковариационную структуру, заданную пользователем и зависящую от расстояния d_{ij} между выборочными точками *i–j*. Чаще всего используется экспоненциальная функция $f(d_{ij}) = \sigma^2_S \exp(-d_{ij}/\alpha)$, где пространственная дисперсия (σ^2_s) и вектор масштаба (α) являются положительными параметрами пространственного случайного эффекта, который оценивается при построении модели.

Если отклоняется гипотеза, что все виды в сообществе функционируют независимо, то совокупность статистически значимых положительных или отрицательных взаимодействий между видами может в конечном итоге влиять на индивидуальную численность **Y** каждого из них. По этой причине целесообразно включить в многомерный анализ случайный эффект, учитывающий дополнительную информацию о том, какие виды встречаются совместно "чаще, чем случайно". Последняя фраза означает одновременное присутствие пары видов на *i*-м участке с вероятностью, превышающей ту, которая ожидается из сходства параметров $\beta_{i.}$ их ниш. В матричной форме эффект ассоциативности видов записывается как $L_{i.}^{R} \sim N(0, \mathbf{\Omega})$, где $\mathbf{\Omega} = \mathbf{\Lambda}^T\mathbf{\Lambda}$ –экологически ограниченная корреляционная матрица видов. Таким образом, эта группа случайных эффектов генерирует остаточные ковариации сверх тех, что учтены фиксированными эффектами, т.е. выделяет только те ассоциации, которые не могут быть объяснены экологическими ковариатами x_{ik}, уже включенными в модель.

6.3. Модели пространственного распределения одного вида

Построение одномерных моделей HMSC распределения численности видов и таксономических групп макрозообентоса рассмотрим на примере подсемейства Prodiamesinae (Diptera, Chironomidae), все виды которого условно принимались экологически идентичными, а их численности суммировались и логарифмировалась. Всего виды этого таксона были обнаружены на 41 участке рек из 132 обследованных.

Модели HMSC будем строить на основе 7 непрерывных переменных или ковариат: геофизических (высота `Alt` и индекс шероховатости рельефа `TRI`) и климатических (среднегодовая температура `MTemp = BIO1` и объем осадков самого жаркого квартала `PrecDQ = BIO17`) переменных и показателей качества воды (`Miner,NH4,O2`). Географические координаты участков и категории типов грунтов дна рек интерпретируем как случайные факторы `Rivers` и `Ground` соответственно.

```
# Можно прологарифмировать отклик и так
library(vegan)
yp <- decostand(df$PRODIAMESINAE, method="log")
Y2 <- as.matrix(as.data.frame(yp))
#   Подготавливаем таблицу из 8 независимых переменных
clipok <- df.clim[,c(1,17,20,21)]
colnames(clipok) <- c("MTemp","PrecDQ", "Alt", "TRI")
df_m <- cbind(clipok, df_var[,-1])
```

Первую модель только из одних фиксированных получим, используя 7 ковариат. Апостериорное распределение коэффициентов модели получим с использованием марковского процесса из 30000 итераций для 4 цепей Монте-Карло (`nChains`). Первые 10000 итераций отбрасывались, поскольку еще не прошла стабилизация значений. Большое значение для оценки продолжительности псевдо-случайной имитации МСМС имеют число цепей и величина интервала "истончения" (`Thinning`).

```
library(Hmsc)
# ---------------  Модель 1 ------------------
m1_f = Hmsc(Y = Y2, XData = df_m,
                XFormula = ~MTemp+PrecDQ+Alt+TRI+Miner+NH4+O2)
thin = 20
samples = 1000
nChains = 4
transient = 500*thin
verbose = 500*thin
m1_f = sampleMcmc(m1_f, thin = thin, samples = samples,
       transient = transient, nChains = nChains, verbose = verbose)
Hmsc object with 132 sampling units, 1 species, 8 covariates, 1 traits and 0 random levels
Posterior MCMC sampling with 4 chains each with 1000 samples, thin 20 and transient 10000
```

Построенная апостериорная модель конвертируется в набор объектов `Coda` типа `list`, которые включают всю информацию об оцененных параметрах.

```
mpost1_f = convertToCodaObject(m1_f)
summary(mpost1_f$Beta)
Iterations = 10020:30000
Thinning interval = 20
Number of chains = 4
Sample size per chain = 1000

1. Empirical mean and standard deviation for each variable,
   plus standard error of the mean:
                                   Mean        SD  Naive SE Time-series SE
B[(Intercept) (C1), V1 (S1)]  4.707e+00 6.756e+00 1.068e-01      1.068e-01
B[MTemp (C2), V1 (S1)]       -6.000e-02 5.971e-02 9.441e-04      9.438e-04
B[PrecDQ (C3), V1 (S1)]      -1.357e-02 4.555e-02 7.202e-04      7.299e-04
B[Alt (C4), V1 (S1)]          2.654e-02 9.237e-03 1.460e-04      1.482e-04
B[TRI (C5), V1 (S1)]          1.342e-02 9.536e-03 1.508e-04      1.508e-04
B[Miner (C6), V1 (S1)]        2.349e-05 6.252e-05 9.885e-07      9.989e-07
B[NH4 (C7), V1 (S1)]          4.541e-02 1.199e-01 1.896e-03      1.886e-03
B[O2 (C8), V1 (S1)]          -9.950e-03 1.458e-02 2.305e-04      2.259e-04

2. Quantiles for each variable:
                                 2.5%        25%        50%        75%     97.5%
B[(Intercept) (C1), V1 (S1)] -8.4448720  2.040e-01  4.750e+00  9.241e+00 1.797e+01
B[MTemp (C2), V1 (S1)]       -0.1755918 -1.008e-01 -5.910e-02 -2.050e-02 5.543e-02
B[PrecDQ (C3), V1 (S1)]      -0.1037535 -4.305e-02 -1.334e-02  1.709e-02 7.319e-02
B[Alt (C4), V1 (S1)]          0.0084474  2.043e-02  2.685e-02  3.260e-02 4.456e-02
B[TRI (C5), V1 (S1)]         -0.0051815  6.896e-03  1.358e-02  1.981e-02 3.249e-02
B[Miner (C6), V1 (S1)]       -0.0001003 -1.909e-05  2.313e-05  6.725e-05 1.438e-04
B[NH4 (C7), V1 (S1)]         -0.1871678 -3.409e-02  4.522e-02  1.267e-01 2.830e-01
B[O2 (C8), V1 (S1)]          -0.0387716 -1.963e-02 -1.004e-02 -1.461e-04 1.820e-02
```

Оценить, насколько построенная модель соответствует исходным данным, можно с применением тех же оценок качества, что и в разделе 5, т.е. корня из среднеквадратичной ошибки `RMSE` и квадрата коэффициента детерминации `R2`. Можно также использовать какой-нибудь из семейства информационных критериев – AIC, BIC или "широко применимый" AIC (WAIC – Widely Applicable Bayesian Information Criterion, Watanabe, 2010):

```
preds = computePredictedValues(m1_f)
evaluateModelFit(hM=m1_f, predY=preds)
$RMSE [1] 3.99032
$R2   [1] 0.287871
(WAIC = computeWAIC(m1_f))
[1] 4283.804
```

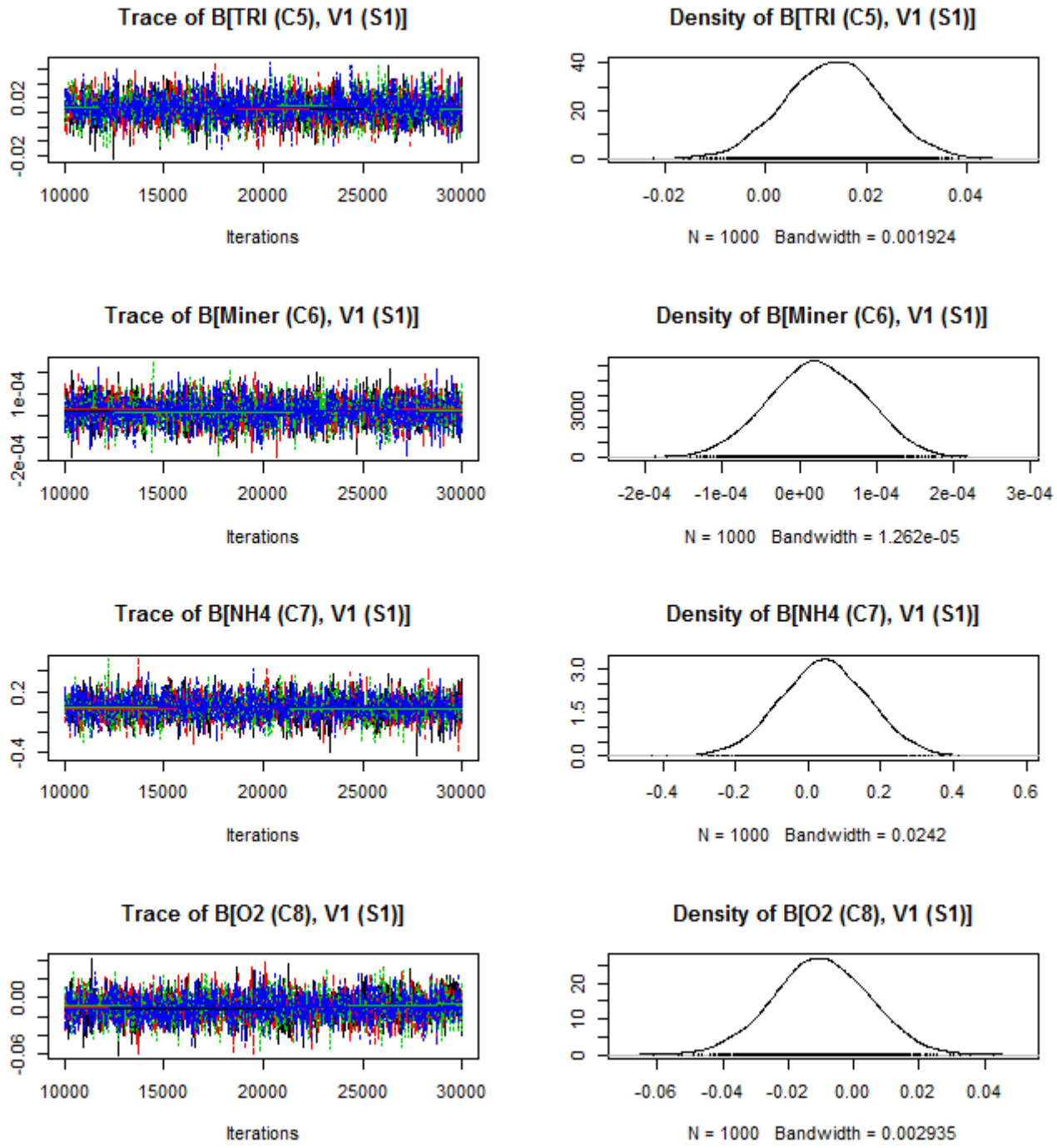
Поскольку в приведенном случае предсказываются те же данные, что были использованы для подгонки модели, поэтому R^2 оценивает объясняющую, а не

прогнозирующую силу модели. Получить независимую эффективность предсказания для каждой единицы выборки можно с помощью перекрестной проверки. Как и в предыдущем разделе выполним скользящий контроль с разбиением на пять групп:

```
Part <- createPartition(m1_f, nfolds =5)
preds = computePredictedValues(m1_f, partition = Part)
evaluateModelFit(hM = m1_f, predY = preds)
$RMSE [1] 4.237257
$R2   [1] 0.2034329
```

Важным моментом является диагностика достаточности имитационного процесса и проверка исходных предпосылок модели. Здесь представляют интерес графики трассировки сходимости МСМС и распределений оценок параметров:

```
plot(mpost1_f$Beta)
```



Если имитационный процесс удачно завершен, то: а) все цепи МСМС дают практически идентичные результаты, б) цепи хорошо перемешаны и флуктуируют без какой-либо видимой автокорреляции и в) они достигли стационарного распределения (левая половина сохраненных итераций выглядит статистически идентично правой половине).

Степень конвергенции цепей МСМС можно также оценить количественно с точки зрения критерия эффективного размера выборки и коэффициента потенциального уменьшения масштаба.

```
data.frame(effectiveSize(mpost1_f$Beta))
                                effectiveSize.mpost1_f.Beta.
B[(Intercept) (C1), V1 (S1)]                        4000.000
B[MTemp (C2), V1 (S1)]                              4000.000
B[PrecDQ (C3), V1 (S1)]                             3900.372
B[Alt (C4), V1 (S1)]                                3911.241
B[TRI (C5), V1 (S1)]                                4000.000
B[Miner (C6), V1 (S1)]                              3921.570
B[NH4 (C7), V1 (S1)]                                4043.098
B[O2 (C8), V1 (S1)]                                 4174.985
gelman.diag(mpost1_f$Beta, multivariate=FALSE)$psrf
                                Point est. Upper C.I.
B[(Intercept) (C1), V1 (S1)]  1.0013115   1.005426
B[MTemp (C2), V1 (S1)]        1.0017297   1.006560
B[PrecDQ (C3), V1 (S1)]       1.0006060   1.003207
B[Alt (C4), V1 (S1)]          1.0008446   1.003573
B[TRI (C5), V1 (S1)]          1.0005996   1.002753
B[Miner (C6), V1 (S1)]        1.0000775   1.001145
B[NH4 (C7), V1 (S1)]          1.0013080   1.004357
B[O2 (C8), V1 (S1)]           0.9998985   1.000247
```

Заметим, что эффективные размеры выборок значительно меньше фактического числа выборок (30000), т.е. имитация выполнена с запасом. Коэффициенты потенциального уменьшения масштаба очень близки к единице, что указывает на близость результатов, которые дают все четыре цепи Монте-Карло.

Наконец, представляет значительный интерес относительная важность каждого показателя, использованного для прогнозирования величины отклика, которая оценивается по их доле в разложении общей объясненной дисперсии по всем фиксированным и случайным факторам.

```
# по отдельным переменным
VP <- computeVariancePartitioning(m1_f)
VP$vals
               V1
MTemp  0.21204008
PrecDQ 0.06604424
Alt    0.48982103
TRI    0.09993934
Miner  0.04164172
NH4    0.04149467
O2     0.04901892
# по группам переменных
computeVariancePartitioning(m1_f, group=c(1,2,2,2,2,3,3,3),
             groupnames=c("Inter","Климат","Биотоп"))$vals
               V1
Inter  0.00000000
Климат 0.93238787
Биотоп 0.06761213
```

Вторую модель m2_fr построим на основе факторов, характеризующих условия среды в биотопе фиксированных Miner, NH4, O2 и случайного Ground.

```
# ---------------  Модель 2 ------------------
# Определение случайного фактора для включения в модель
Ground = as.factor(df_m$Ground)
studyDesign = data.frame(Ground  = Ground)
rLGround = HmscRandomLevel(units = studyDesign$Ground)

m2_fr = Hmsc(Y = Y2, XData = df_m, XFormula = ~Miner+NH4+O2,
     studyDesign=studyDesign, ranLevels=list("Ground"=rLGround))
m2_fr = sampleMcmc(m2_fr, thin = thin, samples = samples,
     transient = transient, nChains = nChains, verbose = verbose)
mpost2_fr = convertToCodaObject(m2_fr)
# Коэффициенты бета построенной модели
summary(mpost2_fr$Beta)$statistics
                                   Mean           SD     Naive SE Time-series SE
B[(Intercept) (C1), V1 (S1)]  4.655954e+00 1.592617e+00 2.518149e-02   2.518674e-02
B[Miner (C2), V1 (S1)]       -3.826723e-05 6.894812e-05 1.090165e-06   1.076368e-06
B[NH4 (C3), V1 (S1)]         -3.291595e-02 1.329188e-01 2.101631e-03   2.100625e-03
B[O2 (C4), V1 (S1)]          -1.951980e-02 1.555819e-02 2.459966e-04   2.460327e-04
# Оценки качества приближения модели
preds = computePredictedValues(m2_fr)
evaluateModelFit(hM=m2_fr, predY=preds)
$RMSE [1] 4.488683
$R2   [1] 0.1146387
(WAIC = computeWAIC(m2_fr))
[1] 10865.28
# Оценки важности предикторов модели
VP <- computeVariancePartitioning(m2_fr)
VP$vals
                      V1
Miner          0.2075504
NH4            0.1823863
O2             0.3386337
Random: Ground 0.2714297
```

Модель 2 существенно уступает модели 1 по критериям качества аппроксимации данных RMSE и R2.

Третью модель m3_fr построим с использованием климатических (MTemp, PrecDQ) и геофизических (Alt, TRI) фиксированных факторов, а также случайного фактора Rivers, определяющую пространственную автокорреляционную зависимость.

```
# ---------------  Модель 3 ------------------
# Определение случайного фактора для включения в модель
xy <- as.matrix(df[,c("X","Y")])
studyDesign = data.frame(Rivers = df$NamRiver)
rownames(xy) = studyDesign[,1]
rLRivers = HmscRandomLevel(sData = xy)

m3_fr = Hmsc(Y = Y2, XData = df_m, XFormula = ~MTemp+PrecDQ+Alt+TRI,
     studyDesign=studyDesign, ranLevels=list("Rivers"= rLRivers))
# Уменьшим немного число псевдовыборок для имитации
thin = 10
samples = 1000
nChains = 3
transient = 500*thin
verbose = 500*thin
m3_fr = sampleMcmc(m3_fr, thin = thin, samples = samples,
        transient = transient, nChains = nChains, verbose = verbose)
```

```
mpost3_f = convertToCodaObject(m3_fr)
summary(mpost3_f$Beta) $statistics
                                  Mean       SD  Naive SE Time-series SE
B[(Intercept) (C1), V1 (S1)]  3.05776 7.099114 0.1296115      0.1418261
B[MTemp (C2), V1 (S1)]       -0.05544 0.069248 0.0012643      0.0012746
B[PrecDQ (C3), V1 (S1)]      -0.01267 0.051083 0.0009326      0.0009295
B[Alt (C4), V1 (S1)]          0.02387 0.010216 0.0001865      0.0001919
B[TRI (C5), V1 (S1)]          0.01218 0.009765 0.0001783      0.0001767

preds = computePredictedValues(m3_fr)
evaluateModelFit(hM=m3_fr, predY=preds)
WAIC = computeWAIC(m3_fr)
$RMSE  [1]  2.184037
$R2       [1]  0.8544619
computeWAIC(m3_fr)
[1]  9262.482

Part <- createPartition(m3_fr, nfolds =5)
preds = computePredictedValues(m3_fr, partition = Part)
evaluateModelFit(hM = m3_fr, predY = preds)
$RMSE [1] 4.140216
$R2     [1] 0.2466231

VP <- computeVariancePartitioning(m3_fr)
VP$vals
                      V1
MTemp          0.07345663
PrecDQ         0.02451663
Alt            0.11157381
TRI            0.03964883
Random: Rivers 0.75080409
```

Наконец, четвертую модель `m4_fr` построим на основе всех имеющихся факторов.

```
# ---------------  Модель 4 ------------------
# Определение обоих случайных факторов для включения в модель
studyDesign = data.frame(Rivers = df_m$NamRiver,Ground  = Ground)
rLRivers = HmscRandomLevel(sData = xy)
rLGround = HmscRandomLevel(units = studyDesign$Ground)

m4_fr = Hmsc(Y = Y2, XData = df_m, XFormula =
      ~MTemp+PrecDQ+Alt+TRI+Miner+NH4+O2, studyDesign=studyDesign,
       ranLevels=list("Rivers"= rLRivers, "Ground"=rLGround))

m4_fr = sampleMcmc(m4_fr, thin = thin, samples = samples, transient =
transient,
       nChains = nChains, verbose = verbose)

mpost4_f = convertToCodaObject(m4_fr)
summary(mpost4_f$Beta) $statistics
                                    Mean         SD  Naive SE Time-series SE
B[(Intercept) (C1), V1 (S1)]  3.766e+00 7.053e+00 1.115e-01      1.114e-01
B[MTemp (C2), V1 (S1)]       -5.319e-02 6.850e-02 1.083e-03      1.119e-03
B[PrecDQ (C3), V1 (S1)]      -1.478e-02 5.135e-02 8.119e-04      7.973e-04
B[Alt (C4), V1 (S1)]          2.462e-02 9.849e-03 1.557e-04      1.649e-04
B[TRI (C5), V1 (S1)]          1.156e-02 9.597e-03 1.517e-04      1.582e-04
B[Miner (C6), V1 (S1)]        1.477e-05 6.118e-05 9.674e-07      9.652e-07
B[NH4 (C7), V1 (S1)]          2.873e-02 1.164e-01 1.840e-03      1.840e-03
B[O2 (C8), V1 (S1)]          -8.900e-03 1.431e-02 2.263e-04      2.291e-04

preds = computePredictedValues(m4_fr)
```

```
evaluateModelFit(hM=m4_fr, predY=preds)
$RMSE [1] 3.111985
$R2     [1] 0.6132854
computeWAIC(m2_fr)
 9634.228

VP <- computeVariancePartitioning(m4_fr)
VP$vals
                        V1
MTemp          0.121635388
PrecDQ         0.043584523
Alt            0.221391653
TRI            0.051765604
Miner          0.022407285
NH4            0.019877357
O2             0.024106378
Random: Rivers 0.486797340
Random: Ground 0.008434472

computeVariancePartitioning(m4_fr, group=c(1,2,2,2,2,3,3,3),
                 groupnames=c("Inter","Климат","Биотоп"))$vals
                        V1
Inter          0.000000000
Климат         0.460128295
Биотоп         0.044639893
Random: Rivers 0.486797340
Random: Ground 0.008434472
```

Сравнительный анализ коэффициентов моделей HMSC позволяет установить приоритеты внешних факторов по степени их влияния на пространственное распределение популяционной плотности видов. В частности, представленные результаты свидетельствуют о сильной зависимости численности Prodiamesinae от картографических координат и высоты местности над уровнем моря. Это обычно характерно для ареалов, ограничивающих четко выраженный географический кластер. Действительно, личинки этого подсемейства являются выраженными рео- и оксибионтами и обитают на каменно-песчаных биотопах проточных рек арктоальпийского типа, которые чаще встречаются на северо-востоке региона.

Из четырех построенных моделей наилучшая объясняющая способность – у модели 3. Однако кросс-проверка показала довольно низкий уровень прогноза на независимой выборке. Модель 4 на полном наборе параметров оказалась несколько хуже модели 3, что свидетельствует, вероятно, о ее переобучении.

Попробуем теперь получить прогнозируемые значения численности продиамезин для исходной выборки 132 рек. Однако в пакете `Hmsc` мы не нашли общепринятой функции `predict()`. В частности, функция `computePredictedValues()` возвращает матрицу 132×3000 для каждой из 3000 итераций имитаций:

```
preds = computePredictedValues(m3_fr)
str(preds)
num [1:132, 1, 1:3000] 8.803 2.818 8.019 0.889 -0.152 ...
 - attr(*, "dimnames")=List of 3
  ..$ : chr [1:132] "1" "2" "3" "4" ...
  ..$ : chr "V1"
  ..$ : NULL
```

Используем для осреднения прогнозов следующую подходящую функцию:

```
library(abind)
mPred = function(hM, predY)
{   median2 = function(x) {
        return(median(x, na.rm = TRUE))
    }
    mean2 = function(x) { return(mean(x, na.rm = TRUE))
    }
    mPredY = matrix(NA, nrow = hM$ny, ncol = hM$ns)
    sel = hM$distr[, 1] == 3
    if (sum(sel) > 0) {
        mPredY[, sel] = as.matrix(apply(abind(predY[, sel, ,
            drop = FALSE], along = 3), c(1, 2), median2))
    }
    sel = !hM$distr[, 1] == 3
    if (sum(sel) > 0) {
        mPredY[, sel] = as.matrix(apply(abind(predY[, sel, ,
            drop = FALSE], along = 3), c(1, 2), mean2))
    }
  mPredY
}
YP <- mPred(hM=m3_fr, predY=preds)
```

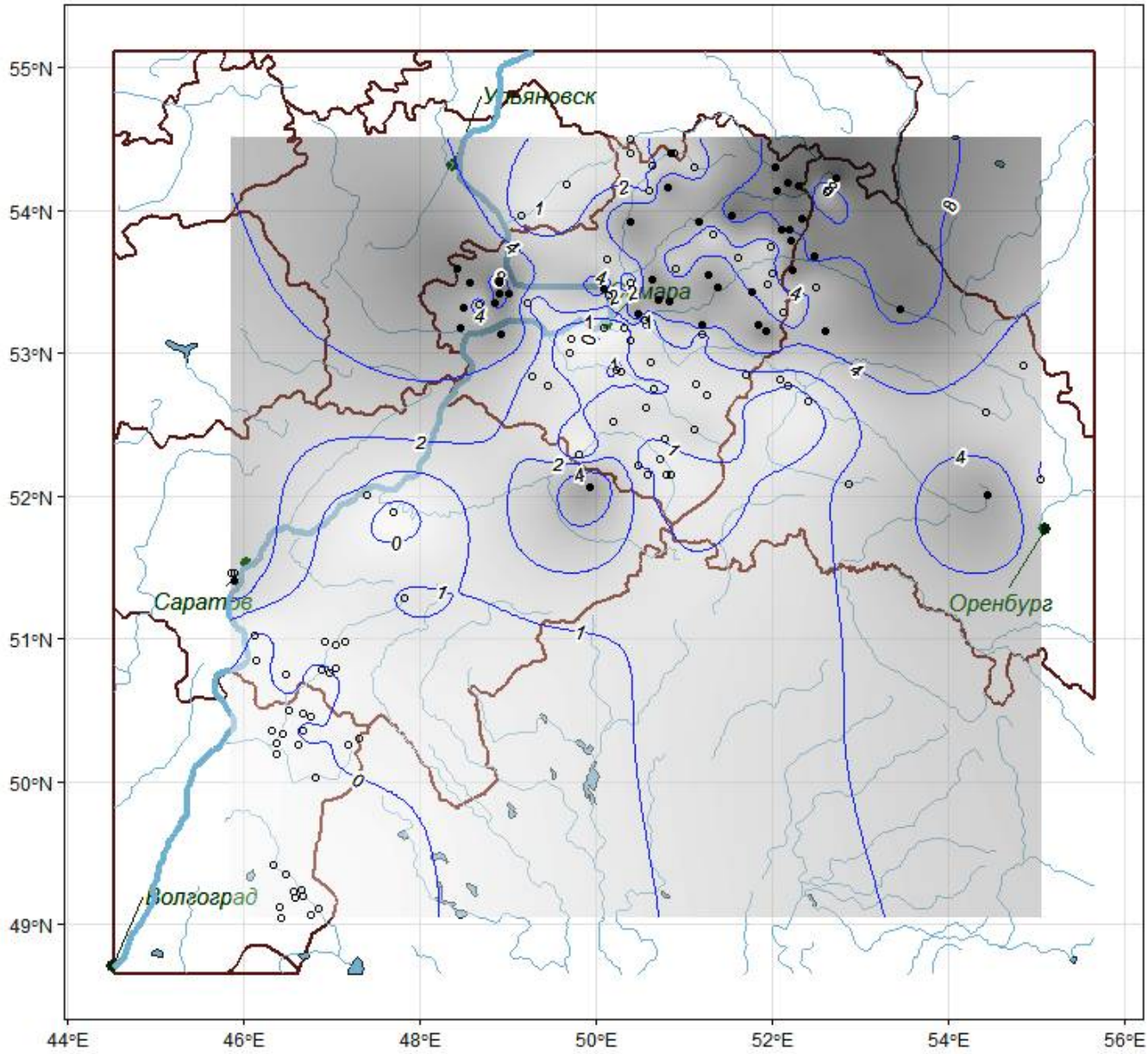
Теперь можно рассчитать остатки модели и провести с ними все общепринятые статистические процедуры (найти выбросы, оценить распределение остатков и их зависимость от отклика и проч.). Но мы ограничимся здесь построением карты пространственного распределения прогнозируемой численности подсемейства.

```
toMap <- df[,c("X","Y")]
toMap$YP <- YP
toMap$Y2 <-  Y2

# Выполним интерполяцию прогнозируемых значений
library(MBA)                # метод иерархических базисных сплайнов
library(reshape2)
toMap <- as.data.frame(toMap)
mba.int <- mba.surf(toMap[,-4], 300, 300, extend=TRUE)$xyz.est
dimnames(mba.int$z) <- list(mba.int$x, mba.int$y)
mba.output <- melt(mba.int$z, varnames = c('X', 'Y'), value.name = 'Z')
toMap1 <- toMap[toMap$V1 !=0, (1:2)]
toMap0 <- toMap[toMap$V1 ==0, (1:2)]
```

На картосхеме проведения исследований кружками отметим районы взятия проб (черным цветом залиты точки, где были обнаружены Prodiamesinae). Серым цветом разной интенсивности покажем прогнозируемое по модели m3_fr распределение популяционной плотности этого подсемейства. Контурами отметим изолинии прологарифмированной численности (экз/м^2).

```
# Проведем раскраску базовой карты и проведение изолиний
brk <- c(0, 1, 2, 4, 8, 15)
Basemap +
geom_tile(data = mba.output, aes(x = X, y = Y, fill=Z), alpha = 0.5) +
      scale_fill_gradient(low = "white", high = "gray10") +
    geom_contour(data = mba.output, aes(x = X, y = Y, z=Z), breaks = brk) +
       geom_text_contour(data = mba.output,aes(x = X, y = Y, z=Z),
                   stroke=0.2, breaks = brk, fontface = "italic", size=3) +
        geom_point(data=toMap0, aes(x=X, y=Y), shape=21) +
        geom_point(data=toMap1, aes(x=X, y=Y)) +
        theme_bw()
```



7. Модели совместного распределения ансамбля видов

Многомерную модель HMSC рассмотрим на примере оценки пространственного распределения сообщества из 31 вида личинок хирономид. Для того, чтобы реализовать построение модели, необходимо с уже знакомого читателю общедоступного ресурса скачать файл http://www.ievbras.ru/ecostat/Kiril/R/Blog/Chir_spec.RData , в котором приведена вся необходимая информация. Таблица `TB_Ch.wide` содержит данные о средней численности каждого вида хирономид, привязанные к географическим координатам тех же 132 малых рек, где выполнялись гидробиологические пробы. Таблица `Species_Tax31` по каждому из этих видов содержит информацию для построения филогенетического дерева.

Для численностей видов разумно предварительно выполнить нормализующее преобразование, приводящее к χ^2-дистанции, которое является, по всей вероятности, наиболее разумным компромиссом при учёте как роли ведущих компонент, так и вклада редких или малочисленных таксонов (Legendre, Gallagher 2001).

```
load(file="Chir_spec")

ibrary(vegan)
# Преобразование значений численностей
YCh31 <- decostand(TB_Ch.wide[,-(1:2)], method="chi.square")

library(picante)
# Построение таксономического дерева
taxdis <- vegan::taxa2dist(Species_Tax31[,3:6], varstep = TRUE)
spe.ch <- hclust(taxdis, method="complete")
library(ape)
PTree31 <- as.phylo(spe.ch)
plot(PTree31)
```

В качестве предикторов модели используем те же переменные, что и в предыдущих примерах для модели одного вида с добавлением матрицы филогенетических корреляций **C**. Характеристики биотопов, выраженные категориальной переменной `Ground` (от 1 – чистый песок или галька до 6 – черный ил и растительные остатки), в этот раз будем интерпретировать как фиксированный фактор. Построим две модели, из которых вторая будет включать дополнительно случайный фактор `Rivers`, определяющий пространственную автокорреляционную зависимость.

```
# ---------------  Модель 1 ------------------
mmn =  Hmsc(Y=as.matrix(YCh31), XData= df_m,
    XFormula = ~MTemp+PrecDQ+Alt+TRI+Miner+NH4+O2+Ground,
    phyloTree = PTree31)
mmn = sampleMcmc(mmn, thin = thin, samples = samples,
    transient = transient, nChains = nChains, verbose = verbose)
preds = computePredictedValues(mmn)
MF = evaluateModelFit(hM = mmn, predY = preds)
$RMSE
 [1] 0.3962001 0.5240846 0.5622915 0.1656079 0.7028886 0.6153581 0.2152136 0.3941166
 [9] 0.3191814 0.4907065 0.1559568 0.1724076 0.5129015 0.5013489 0.4912206 0.3124545
[17] 0.4892231 0.4013550 0.4492891 0.4789234 0.6978103 0.4499466 0.5767323 0.4495444
[25] 0.4389911 0.6456889 0.4864389 0.4209956 0.2381295 0.6021028 0.4517233
$R2
 [1] 0.03579749 0.19717332 0.02956938 0.74848874 0.02555792 0.12504801 0.78933439
 [8] 0.07083157 0.12215600 0.03118024 0.30259311 0.34427553 0.09441079 0.09697804
[15] 0.09927049 0.05373104 0.08314298 0.14408006 0.14350363 0.12781503 0.03750803
[22] 0.11779930 0.14334937 0.20114494 0.07272065 0.08535401 0.06736052 0.26618083
[29] 0.25125175 0.04076763 0.13649768
mean(MF$R2)
[1] 0.1640281

# ---------------  Модель 2 ------------------
# Определение случайного фактора для включения в модель
xy <- as.matrix(df[,c("X","Y")])
studyDesign = data.frame(Rivers = df$NamRiver)
rownames(xy) = studyDesign[,1]
rLRivers = HmscRandomLevel(sData = xy)

mnn_r = Hmsc(Y=as.matrix(YCh31), XData= df_m,
     XFormula = ~MTemp+PrecDQ+Alt+TRI+Miner+NH4+O2+Ground,
     phyloTree = PTree31,
     studyDesign=studyDesign, ranLevels=list("Rivers"= rLRivers))
mnn_r <- sampleMcmc(mnn_r, thin = thin, samples = samples,
     transient = transient, nChains = nChains, verbose = verbose)

preds = computePredictedValues(mmn)
MF = evaluateModelFit(hM = mmn, predY = preds)
```

```
$RMSE
 [1] 0.39269335 0.50975766 0.56157260 0.04148669 0.70139378 0.61488903 0.09683960
 [8] 0.39352464 0.31457532 0.48998990 0.07732134 0.03926303 0.51211605 0.49974587
[15] 0.49075614 0.31207991 0.48537892 0.40036421 0.44884774 0.46382487 0.69713793
[22] 0.44960602 0.57383216 0.44322341 0.43864678 0.64504953 0.47941733 0.41545570
[29] 0.22538090 0.60145440 0.44743887
$R2
 [1] 0.05580129 0.24336595 0.03262072 0.98518783 0.03114058 0.12651990 0.95803763
 [8] 0.07416681 0.14807896 0.03366905 0.83365526 0.97053204 0.09682629 0.10343203
[15] 0.10098063 0.05604728 0.09841060 0.14841838 0.14547679 0.18228151 0.04066904
[22] 0.11963179 0.15238120 0.22441873 0.07426588 0.08724356 0.09691372 0.28741570
[29] 0.33031484 0.04284037 0.15399909
mean(MF$R2)
[1] 0.2269272
```

Поскольку для любого из видов модель 2 имеет лучшие характеристики RMSE и R2, дальнейший анализ будем проводить только на ее основе.

```
# Оценки важности групп переменных для каждого вида
VP = computeVariancePartitioning(m4_fr, group=c(1,2,2,2,2,3,3,4),
                 groupnames=c("Inter","Климат","Биотоп"","Грунт"))$vals
t(VP$vals[-1,])
             Климат    Биотоп      Грунт   Random: Rivers
ChAbl.m. 0.32334154 0.23963523 0.070051264      0.3669720
ChChi.ap 0.30478386 0.11944302 0.156986757      0.4187864
ChChi.o. 0.35728536 0.24685063 0.115139183      0.2807248
ChChi.sr 0.01015075 0.75258171 0.001981421      0.2352861
ChCor.s. 0.40162637 0.26708812 0.093494397      0.2377911
ChCri.b. 0.54598923 0.20723009 0.110308631      0.1364721
ChCri.sf 0.02043957 0.48316420 0.002406842      0.4939894
ChDic.n. 0.49358028 0.20437141 0.095937660      0.2061107
ChEch.a. 0.51189763 0.14413014 0.026709370      0.3172629
ChGly.g. 0.38236517 0.29043613 0.075100907      0.2520978
ChGly.sl 0.04139779 0.09050359 0.006357051      0.8617416
ChMch.d. 0.03555166 0.11180909 0.007920784      0.8447185
ChMch.t. 0.54396149 0.20248409 0.058130224      0.1954242
ChMit.p. 0.44140221 0.16986541 0.197001860      0.1917305
ChMnd.ba 0.59221156 0.13641211 0.121784696      0.1495916
ChPat.a. 0.51458120 0.19321143 0.060406548      0.2318008
ChPchvr. 0.47644331 0.21747272 0.037260506      0.2688235
ChPlc.co 0.64316970 0.12289361 0.095194162      0.1387425
ChPol.b. 0.56296510 0.17124324 0.103414065      0.1623776
ChPol.n. 0.24106138 0.16411166 0.025875252      0.5689517
ChPol.s. 0.38154485 0.30720845 0.083990435      0.2272563
ChPrc.f. 0.22832288 0.54058323 0.059005226      0.1720887
ChPro.o. 0.57583600 0.13143348 0.148456108      0.1442744
ChPse.s. 0.54861858 0.19579929 0.017396248      0.2381859
ChPtt.co 0.49893350 0.19938098 0.111186427      0.1904991
ChRhe.f. 0.63604729 0.16977599 0.051904770      0.1422719
ChSchcs. 0.23448321 0.20140495 0.198016498      0.3660953
ChTan.p. 0.55390709 0.19494245 0.023087794      0.2280627
ChTar.kr 0.14957451 0.21982881 0.037219363      0.5933773
ChTar.p. 0.31908135 0.35402859 0.071325500      0.2555646
ChTar.sp 0.39774482 0.23175332 0.040140518      0.3303613

mpost = convertToCodaObject(mnn_r)
summary(mpost$Rho)$statistics
          Mean             SD       Naive SE Time-series SE
  0.9912933333   0.0130044990   0.0002374286   0.0002373776
```

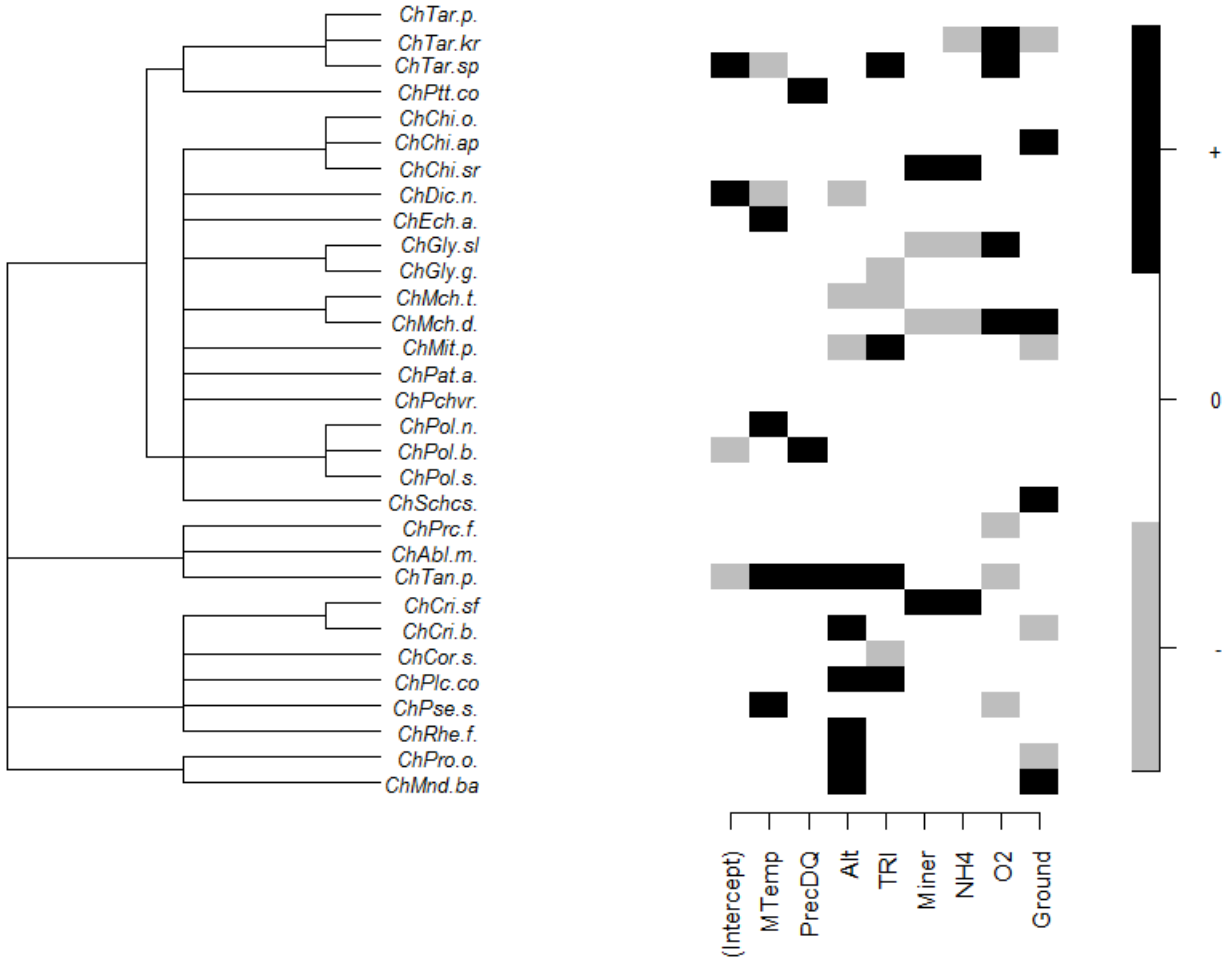
Филогенетический сигнал ρ имеет апостериорное распределение со средним 0.991±0.00024, что доставляет убедительные доказательства весьма высокого влияния таксономической иерархии при выделении экологических ниш.

```
summary(mpost$Alpha[[1]])$statistics
                        Mean         SD     Naive SE Time-series SE
Alpha1[factor1] 2.972839594 2.36601773 0.0431973761    0.290460574
Alpha1[factor2] 0.006228637 0.02621665 0.0004786484    0.003173467
Alpha1[factor3] 3.344956182 3.61345309 0.0659723257    0.657322931
Alpha1[factor4] 2.359016288 3.34219795 0.0610199070    0.556504239
Alpha1[factor5] 0.219034479 1.20913457 0.0220756760    0.173889875
```

Вектор пространственного масштабирующего фактора α имеет характерную пульсирующую последовательность значений со средними $\alpha_1 = 2.97$, $\alpha_2 = 0.006$, $\alpha_3 = 3.34$, $\alpha_4 = 2.35$, $\alpha_5 = 0.22$, разумного объяснения скрытого смысла которой нам найти пока не удалось.

Коэффициенты β удобнее представить в графическом виде

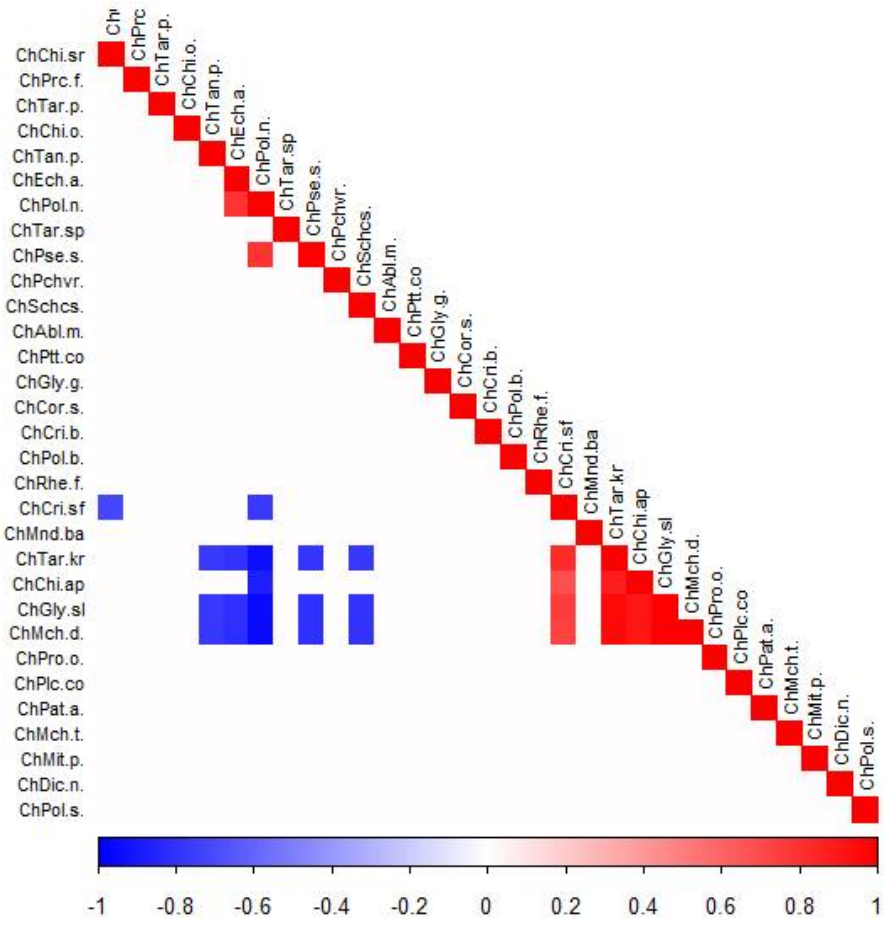
```
postBeta = getPostEstimate(mnn_r, parName = "Beta")
plotBeta(mnn_r, post = postBeta, param = "Sign", plotTree = TRUE,
             colors = colorRampPalette(c("gray", "white","black" )),
             cex = c(0.9, 0.8, 0.8),
             supportLevel = 0.90, split = 0.4, spNamesNumbers = c(T,F),
             covNamesNumbers= c(T,F))
```



На графике черным цветом отмечены ячейки для видов, апостериорное распределение коэффициентов которых статистически значимо смещено в положительную область, т.е. в сторону увеличения соответствующего предиктора. Серым цветом отмечена обратная ситуация, когда снижение значения независимой переменной приводит к увеличению численности видов.

Матрица **Ω** определяет остаточные ковариационные связи между рассматриваемыми видами и ее можно представить в виде корреляционной матрицы:

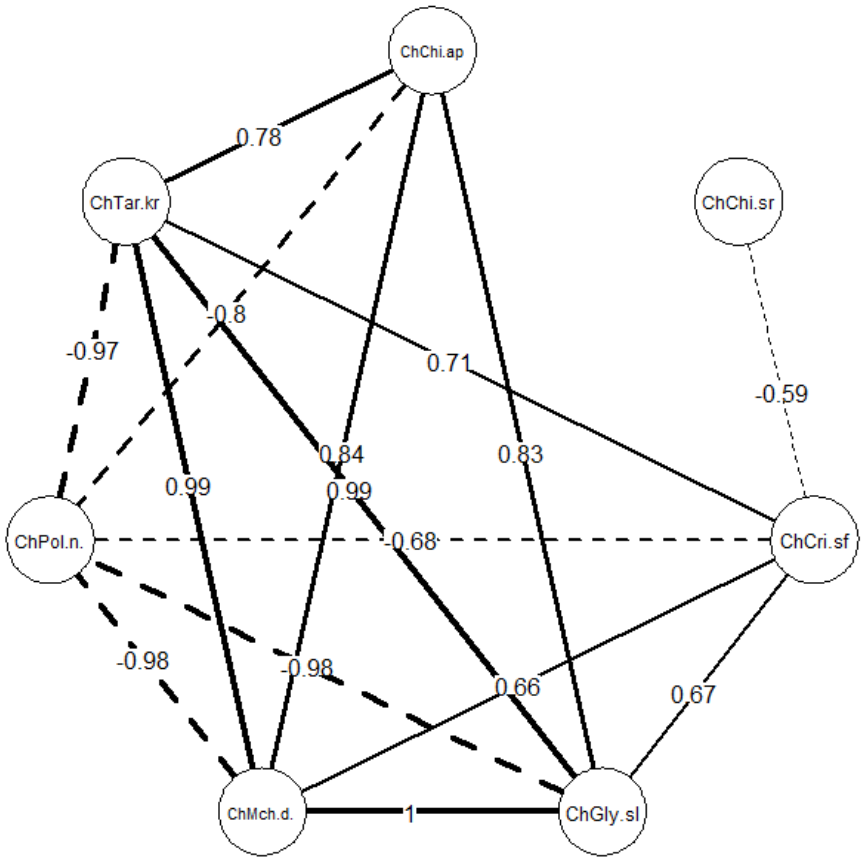
```
library(corrplot)
OmegaCor = computeAssociations(mnn_r)
supportLevel = 0.95
plotOrder = corrMatOrder(OmegaCor[[1]]$mean,order="AOE")
toPlot = ((OmegaCor[[1]]$support>supportLevel) +
       (OmegaCor[[1]]$support<(1-supportLevel))>0)*OmegaCor[[1]]$mean
colnames(toPlot)=rownames(toPlot)=gsub("_"," ",x=colnames(toPlot))
corrplot(toPlot[plotOrder,plotOrder], method = "color",
   col=colorRampPalette(c("blue","white","red"))(200),
   title="",type="lower",tl.col="black",tl.cex=.7, mar=c(0,0,6,0))
```



Возможно, удобнее представить ту же матрицу в виде корреляционной сети (см. https://stok1946.blogspot.com/2020/01/blog-post.html)

```
library(qgraph)
head(toPlot)
# Убираем из матрицы виды, которые не связаны ни с одним другим
i_keep <- c()
for (col in colnames(toPlot)) {
  if (sum(toPlot[,col]) !=1 ) {
    i_keep <- c(i_keep, col)
  }
}
Cor_P <- toPlot[i_keep,i_keep]
```

```
corGraph <- qgraph(Cor_P, graph = "cor", layout = " circular ",
         labels = colnames(Cor_P), cut = 0.5,  minimum = 0.5,
        edge.labels=T, theme = 'gray',   esize =5)
```



Литературные ссылки

Зинченко Т.Д., 2011. Эколого-фаунистическая характеристика хирономид (Diptera, Chhironomidae) малых рек бассейна Средней и Нижней Волги (Атлас). Тольятти: Кассандра. 258 с.

Лисовский А.А., Дудов С.В., Оболенская Е.В., 2020. Преимущества и ограничения использования методов экологического моделирования ареалов. 1. Общие подходы // Журн. общ. биологии. Т. 81. № 2. С. 123–134.

Лисовский А.А., Дудов С.В., 2020а. Преимущества и ограничения использования методов экологического моделирования ареалов. 2. MaxEnt // Журн. общ. биологии. Т. 81. № 2. С. 135–146.

Пузаченко Ю.Г. 2004. Математические методы в экологических и географических исследованиях. М: Академия. 416 с.

Шитиков В. К. , Мастицкий С.Э., 2017. Классификация, регрессия и другие алгоритмы Data Mining с использованием R. Электронная книга. 351 с. URL: https://stok1946.blogspot.com (дата обращения 10.10.2020).

Abrego N., Norberg A., Ovaskainen O., 2017. Measuring and predicting the influence of traits on the assembly processes of wood−inhabiting fungi // Journal of Ecology. V. 105, N 4. P. 1070-1081.

Araújo M.B., Anderson R.P., Barbosa A.M., Beale C.M., Dormann C.F. et al., 2019. Standards for distribution models in biodiversity assessments // Science Advances. V. 5, N 1. P. e4858.

Austin M., Belbin L., Meyers J., Doherty M. , Luoto M. 2006. Evaluation of statistical models used for predicting plant species distributions: role of artifcial data and theory. Ecological Modelling. V. 199 P. 197–216.

Bhattacharya A., Dunson D.B., 2011. Sparse Bayesian infinite factor models // Biometrika. V. 98. P. 291-306.

Breiner F.T., Nobis M.P., Bergamini A., Guisan A., 2018. Optimizing ensembles of small models for predicting the distribution of species with few occurrences // Methods Ecol Evol. V. 9. P. 802-808.

Brooker R.W., 2006. Plant-plant interactions and environmental change // New Phytologist. V. 171. P. 271–284.

Busby J.R., 1991. BIOCLIM – a bioclimate analysis and prediction system // Plant Protection Quarterly. V. 6. P. 8–9.

Calabrese J.M., Certain G., Kraan C., Dormann C.F., 2014. Stacking species distribution models and adjusting bias by linking them to macroecological models // Global Ecology and Biogeography. V. 23. P. 99-112.

Clark J.S., Gelfand A.E., Woodall C.W., Zhu K., 2014. More than the sum of the parts: forest climate response from joint species distribution models // Ecological Applications. V. 24. P. 990-999.

D'Amen M., Rahbek C., Zimmermann N. E., Guisan A., 2017. Spatial predictions at the community level: From current approaches to future frameworks // Biological Reviews. V. 92. P. 169-187.

Franklin J., 2009. Mapping Species Distributions: Spatial Inference and Prediction. Cambridge: Cambridge University Press. 320 p.

Golovatyuk L. V., Shitikov V. K., Zinchenko T. D., 2018. Estimation of the Zonal Distribution of Species of Bottom Communities in Lowland Rivers of the Middle and Lower Volga Basin // Biology Bulletin. V. 45 (10). P. 1262–1268.

Guisan A., Thuiller W., Zimmermann N.E., 2017. Habitat Suitability and Distribution Models:With Applications in R. Cambridge: Cambridge University Press. 478 p.

Harte J. 2011. Maximum Entropy and Ecology: A Theory of Abundance, Distribution, and Energetics. London: Oxford University Press.

Hastie T., Fithian W., 2013. Inference from controversy // Ecography. V. 36. P. 864–867.

HijmansR. J., Cameron S. E., Parra J. L., Jones P. G., Jarvis A. 2005. Very high resolution interpolated climate surfaces for global land areas // International journal of Climatology. V. 25. P. 1965-1978.

Hirzel A. H., Helfer V., Metral F. 2001. Assessing habitat-suitability models with a virtual species // Ecol. Model. V. 145. P. 111–121.

Hutchinson G.E., 1959. Homage to Santa Rosalia or Why are there so many kinds of animals? // Amer. Naturalist. V. 43. № 870. P. 145-159.

Johnson D.H., 1980. The Comparison of Usage and Availability Measurements for Evaluating Resource Preference. Ecology. V. 61, N 1. P. 65-71.

Kearney M.R., 2006. Habitat, environment and niche: What are we modelling? // Oikos. V. 115, N 1. P. 186-191.

Koleff P., Gaston K.J. 2001. Latitudinal gradients in diversity: real patterns and random models // Ecography. V. 24. P. 341–351.

Kursa M., Rudnicki W. 2010. Feature Selection with the Boruta Package // Journal of Statistical Software. V. 036, issue i11

Legendre P., Gallagher E. Ecologically meaningful transformations for ordination of species data // Oecologia. 2001. V. 129. P. 271–280.

Leroy B., Meynard C.N., Bellard C., Courchamp F. 2016. `virtualspecies`: an R package to generate virtual species distributions // Ecography. V. 39. P. 599-607

Manni F., Guerard E., Heyer E. 2004. Geographic patterns of (genetic, morphologic, linguistic) variation: how barriers can be detected by using Monmonier's algorithm // Human Biol. V. 76, № 2. P. 173-190.

Norberg A., Abrego N., Blanchet F.G., Adler F.R., Anderson B.J. et al, 2019.A comprehensive evaluation of predictive performance of 33 species distribution models at species and community levels // Ecological Monographs, . V. 89, N 3. P. e01370.

Ovaskainen O., Abrego N., 2020. Species Distribution Modelling: With Applications in R Cambridge: Cambridge University Press. 370 p.

Ovaskainen O., Abrego, N., Halme, P., Dunson, D., 2016a. Using latent variable models to identify large networks of species-to-species associations at different spatial scales // Methods in Ecology and Evolution. V. 7. P. 549-555.

Ovaskainen O., Roy D.B., Fox R., Anderson B.J., 2016б. Uncovering hidden spatial structure in species communities with spatially explicit joint species distribution models // Methods in Ecology and Evolution. V. 7. P. 428–436.

Ovaskainen O., Tikhonov G., Norberg A., Blanchet F.G., Duan L., Dunson D., Abrego N., 2017. How to make more out of community data? A conceptual framework and its implementation as models and software // Ecology Letters. V. 20. P. 561-576.

Peterson A.T., Soberón J., Pearson R.G., Anderson R.P., Martínez-Meyer E. et al., 2011. Ecological Niches and Geographic Distributions (MPB-49). Princeton:Princeton Univ. Press. 328 p.

Phillips S.J., Anderson R.P., Schapire R.E., 2006. Maximum entropy modeling of species geographic distributions //Ecol. Model. V. 190. № 3–4. P. 231–259.

Pollock L.J., Tingley R., Morris W.K., Golding N., O'Hara R.B. et al., 2014. Understanding co-occurrence by modelling species simultaneously with a Joint Species Distribution Model (JSDM) // Methods in Ecology and Evolution. V. 5. P. 397–406.

Thorson J.T., Ianelli J.N., Larsen E.A., Ries L., Scheuerell M.D. et al., 2016. Joint dynamic species distribution models: a tool for community ordination and spatio-temporal monitoring // Global Ecology and Biogeography. V. 25. P. 1144-1158.

Tikhonov G., Abrego N., Dunson D., Ovaskainen O., 2017. Using joint species distribution models for evaluating how species-to-species associations depend on the environmental context // Methods in Ecology and Evolution. V. 8. P. 443–452.

Tikhonov G., Opedal Ø.H, Abrego N., Lehikoinen A., de Jonge M.J., Oksanen J., Ovaskainen O. et al., 2020. Joint species distribution modelling with the r-package Hmsc // Methods Ecol and Evol. V. 11. P. 442-447.

Vellend M., 2016. The theory of ecological communities. Princeton University Press. 224 p. *Рецензия: Розенберг Г. С., Шитиков В.К., Зинченко Т.Д.*, 2020. // Журн. общ. биологии. Т.. № . С.

Warton D.I., Blanchet F.G., O'Hara R.B., Ovaskainen O., Taskinen S. et al., 2015. So many variables: joint modeling in community ecology //Trends in Ecology and Evolution. V. 30. P. 766-779.

Watanabe S. 2010. Asymptotic Equivalence of Bayes Cross Validation and WidelyApplicable Information Criterion in Singular Learning Theory // Journal of Machine Learning Research. V. 11. P. 3571-3594

Zurell D., Thuiller W., Pagel J., Cabral J.S., Münkemüller T. et al., 2016. Benchmarking novel approaches for modelling species range dynamics // Global Change Biology. V. 22, N 8. P. 2651-2664.