Landscape Resistance
Week 10, Distributed Graduate Seminar 2016

Melanie Murphy
Department of Ecosystem Science & Management
University of Wyoming

Overview

In this activity we will be working with spatial data in R to:

1. Convert spatial data into weights ("costs") based on different weighting approaches.
2. Calculate landscape resistance using least-cost and random walk (analogous to circuit theory) approaches

<u>Note</u> – several steps of this exercise are processor intensive. A couple of the major outputs and an R object are located in the exercise folder. If you are not running 64 bit R, don't attempt to run the circuit theory portions.

Part 1: Importing and exploring data

Raster data is essentially georeferenced matrices (i.e., matrices with geographic coordinates). In this case, we will use spatial data from central Idaho for Columbia spotted frogs (Murphy et al. 2010). Spatial data come from classified Landsat (NLCD), spline-based climate predictions (Rehfeldt et al. 2006), and topographically derived variables (Moore et al. 1993; Evans 1972). Landscape variables were selected based on knowledge of species' ecology and previous research (Murphy et al. 2010; Pilliod et al. 2002; Funk et al. 2005). See Table 1 for descriptions of metrics and brief ecological justifications.

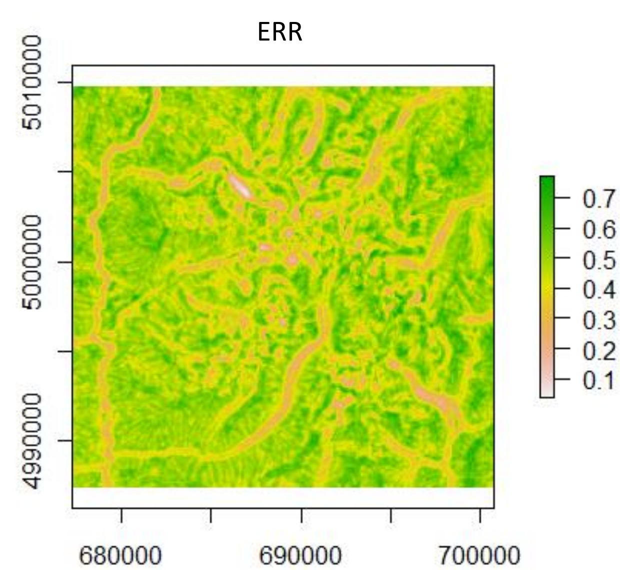
1. Start R and set the working directory (GUI File – Set Directory or setwd() function). The working directory is where all of your data for this exercise are stored.

2. Load the following packages: raster, gdistance, sp, rgdal, igraph, vegan and maptools.

   <u>To load packages:</u>

   require(gdistance)
   require(sp)
   require(rgdal)
   require(igraph)
   require(ecodist)
   require(maptools)

3. Import the rasters (cti, err, ffp and gsp). An .img file is an Imagine file format (a type of raster). If you use ArcMap, you can export as files .img, a file format that is read by a wide variety of programs.



To import the rasters:

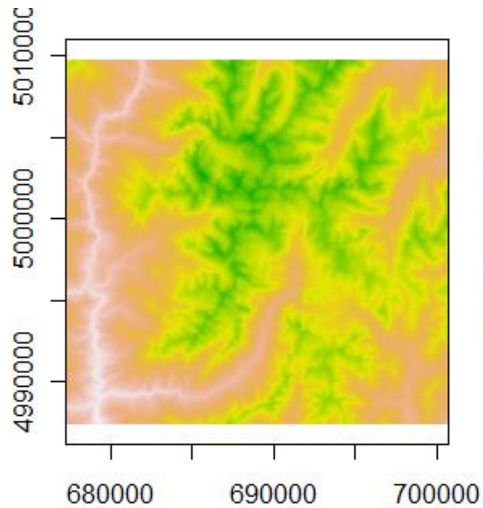
err <- raster("err.img")

repeat this function for cti, ffp, and gps rasters

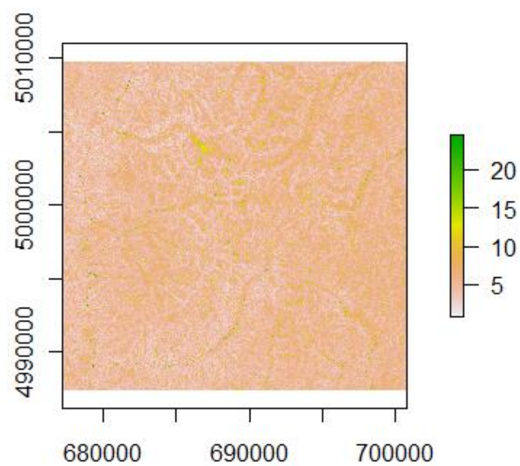
4. Explore these rasters by plotting them. As you go through this exercise, use the plot function at each step to make sure your outputs "make sense".

   For example (see figure):

   plot (err)



GSP



CTI

5. In addition to the spatial data, you also need sample locations. Sample locations are included in the data folder (RALU_UTM.csv). Read in site locations (wetlands with Columbia spotted frogs).

   What are UTMs and why might it be important to work in UTMs (as opposed to lat/longs)?

   sites <- read.csv("RALU_UTM.csv ")

   This reads in the file (*sites* returns the site information contained in this new object). However, R uses a special structure for spatial objects.

6. Next, you will turn your sites object into a spatial points data frame.

   #tell R where the spatial coordinates are in the *sites* object

   xy.dat <- sites[c("X","Y")]

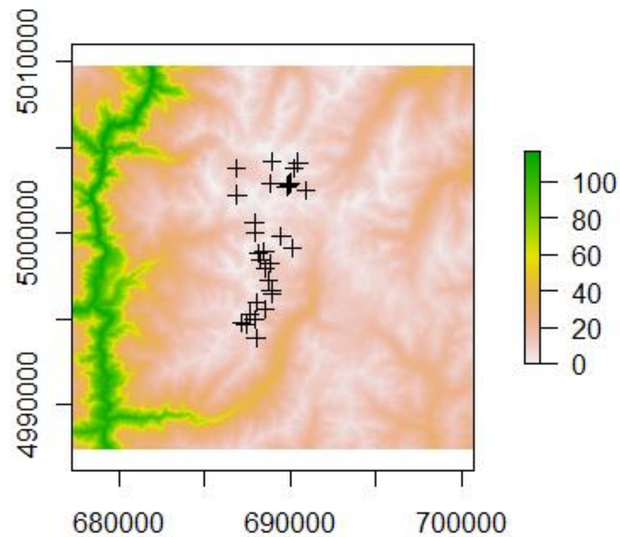
   #create a new *sites* object that is a spatial points data frame with the X Y coordinates

   sites <- SpatialPointsDataFrame(coords=xy.dat, sites)

7. You can now plot your spatial points over your ffp raster. These are the locations with Columbia spotted frog samples within out study area.

   Notice that the area of the raster is much larger than the sample distribution. In the case of ffp, the high values are actually outside of our samples distribution.

   plot(ffp)
   points(sites, pch=3)



Part 2: Setting "costs"

Now you will create categorical data ("costs") from the continuous rasters.

For the purposes of this exercise, we will create a single "landscape resistance" cost surface for each scenario. When analyzing your own data, **you would be advised to test the significance of each landscape variable independently**, even if you then go on to create one landscape resistance surface. You could also test each landscape variable independently in a model selection framework. You will create two sets of total "costs". One will be on "expert opinion" (really more of a "best guess" and not a formal expert opinion) and one will be based on a niche model.

Rescaling the grids will take some processor time (depending on your computer).

You will first use relative ranking of your landscape variables based on expert opinion as follows (i.e., more "weight" the higher the landscape resistance assigned):

Landscape resistances:

1) err, 2) ffp, 3) gsp, 4) cti

There are a couple of important considerations: 1) the transition matrix in gdistance is based on conductance and not resistance and 2) if we are going to create a single landscape resistance (i.e., add the costs together to create one synthetic "landscape" variable), costs values need to represent relative importance of the variables. **Keep in mind there are a multitude of approaches for creating landscape resistance/conductance. This exercise implements one simplistic approach.**

If we flip these values into relative conductance:

1) cti, 2) gsp, 3) ffp, 4) err

Step 1 – get all of the rasters at the same scale

The topographic variables were calculated off a 10 m dem. The climate variables are at 30 m resolution. In order to calculate costs, all of our rasters need to have exactly the same resolution, dimensions, and coordinate locations (i.e., the cells need to match up perfectly).

#resample err and cti (currently at 10 m because they were calculated on a 10 m dem) to dimensions of #gsp (30 m resolution).

```
err <- resample(err, gsp, method= "bilinear")
cti<- resample(cti,gsp,method="bilinear")
```

#Method of resampling (bilinear vs. nearest-neighbor) depends on type of data. Nearest-neighbor is for #categorical data while bilinear interpolation is for continuous data.

Step 2 – Rescale the rasters.

To use the below tools, we need to calculate conductance values. First, look at the range of the variable in a given raster. Then apply a function to get the desired relative conductance values. Make sure to use the plot function and visually inspect your conductance surfaces. Are they as expected?

```
err

class       : RasterLayer
dimensions  : 746, 782, 583372  (nrow, ncol, ncell)
resolution  : 30, 30  (x, y)
extent      : 677292.5, 700762.5, 4987413, 5009783  (xmin, xmax, ymin, ymax)
coord. ref. : +proj=utm +zone=11 +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs
values      : C:\R\Lab8\SpatialData\err.img
min value   : 0.04129123
max value   : 0.7641473
```

Elevation relief ratio (calculated at 27 X 27 raster cell window size) is identifying major topographic features (see Table 1). In this first case, the goal is for err to have the lowest conductance values compared to out other landscape variables. Greater err means more change in topography in a given area. So, the higher the value the more resistance to Columbia spotted frog connectivity. **NOTE**: this gives us a linear relationship between raw value and cost. We could apply any type of functional relationship to a give variable (sigmoidal where low/high values have most resistance, exponential, logarithmic, etc.). I could also normalize all of my variables 0-1 then then apply a cost/conductance to those normalized grids.

```
( err.cost <-(1/err) )

class       : RasterLayer
dimensions  : 746, 782, 583372  (nrow, ncol, ncell)
resolution  : 30, 30  (x, y)
extent      : 677292.5, 700752.5, 4987413, 5009793  (xmin, xmax, ymin, ymax)
coord. ref. : +proj=utm +zone=11 +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs
```

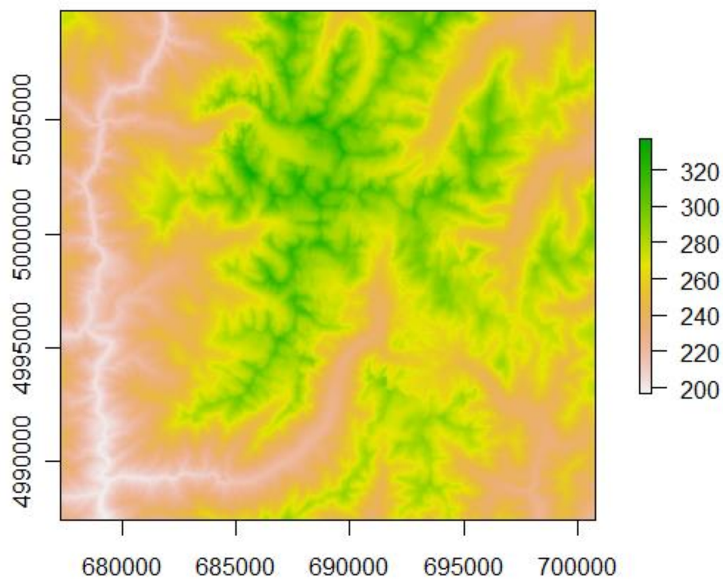
values : in memory
min value : 1.294346
max value : 25.50958

ffp

class : RasterLayer
dimensions : 746, 782, 583372 (nrow, ncol, ncell)
resolution : 30, 30 (x, y)
extent : 677292.5, 700752.5, 4987413, 5009793 (xmin, xmax, ymin, ymax)
coord. ref. : +proj=utm +zone=11 +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs
values : C:\R\Lab8\SpatialData\ffp.img
min value : 0
max value : 117

(ffp.cost <- ffp/5)

class : RasterLayer
dimensions : 746, 782, 583372 (nrow, ncol, ncell)
resolution : 30, 30 (x, y)
extent : 677292.5, 700752.5, 4987413, 5009793 (xmin, xmax, ymin, ymax)
coord. ref. : +proj=utm +zone=11 +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs
values : in memory
min value : 0
max value : 23.4



Note that I set the range a little wide here. Look at the plot of this raster (in relation to the distribution of sample locations).

Question - Why might I choose to have a wider weight range (up to ~23.4) then my target (~1-15)?

gsp

class : RasterLayer
dimensions : 746, 782, 583372 (nrow, ncol, ncell)
resolution : 30, 30 (x, y)
extent : 677292.5, 700752.5, 4987413, 5009793 (xmin, xmax, ymin, ymax)
coord. ref. : +proj=utm +zone=11 +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs
values : C:\R\Lab8\SpatialData\gsp.img
min value : 197
max value : 338.0697

(gsp.cost <- (gsp-196)/15)

Goal – function that inverts max and min values (more growing season precipitation is positively related with connectivity) and rescales ~1-10 in relative cost.

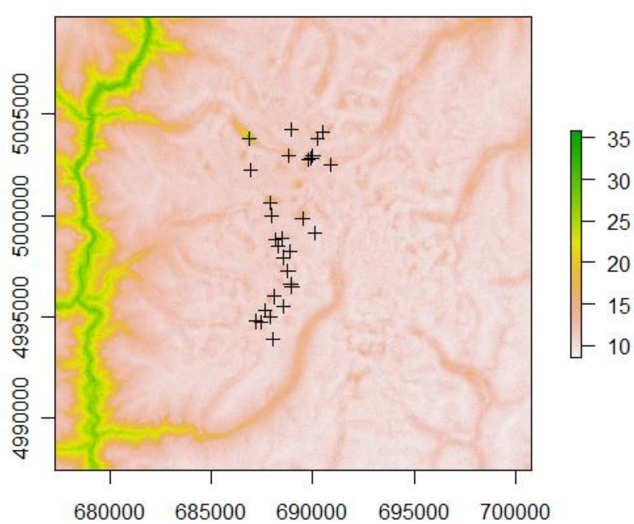
```
class       : RasterLayer
dimensions  : 746, 782, 583372  (nrow, ncol, ncell)
resolution  : 30, 30  (x, y)
extent      : 677292.5, 700752.5, 4987413, 5009793  (xmin, xmax, ymin, ymax)
coord. ref. : +proj=utm +zone=11 +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs
values      : in memory
min value   : 0.06666667
max value   : 9.471311
```

cti

```
class       : RasterLayer
dimensions  : 746, 782, 583372  (nrow, ncol, ncell)
resolution  : 30, 30  (x, y)
extent      : 677292.5, 700752.5, 4987413, 5009793  (xmin, xmax, ymin, ymax)
coord. ref. : +proj=utm +zone=11 +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs
values      : in memory
min value   : 0.897577
max value   : 25.86348
```

(*cti.cost <- cti/5*)

```
class       : RasterLayer
dimensions  : 746, 782, 583372  (nrow, ncol, ncell)
resolution  : 30, 30  (x, y)
extent      : 677292.5, 700752.5, 4987413, 5009793  (xmin, xmax, ymin, ymax)
coord. ref. : +proj=utm +zone=11 +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs
values      : in memory
min value   : 0.1795154
max value   : 5.172697
```



Step 3 – Create a Single Landscape Conductance Raster

In this case, we are going to create a single landscape conductance raster. Remember – testing variables independently may be more appropriate depending on the research question but we will take this approach for the purposes of today's lab.

(cost1 <- (gsp.cost + cti.cost + err.cost + ffp.cost))

class : RasterLayer
dimensions : 746, 782, 583372 (nrow, ncol, ncell)
resolution : 30, 30 (x, y)
extent : 677292.5, 700752.5, 4987413, 5009793 (xmin, xmax, ymin, ymax)
coord. ref. : +proj=utm +zone=11 +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs
values : in memory
min value : 8.569514
max value : 35.95375

Plot your cost surface with your sample locations on top. What does this tell you?

Part 3: Getting from conductance (i.e., 1/cost) to some measure of effective distance.

Step 1 – Creating a Transition Layer

Transition layers are constructed from a raster. The transition layer takes the geographic references (projection, resolution, extent) from the original raster object. It also contains a matrix of probability of movement between cells which can be interpreted as "conductance". Each cell in the matrix represents a cell in the original raster object.

The first step is to construct a transition object based on our "cost1" (which is really conductance as calculated). This step is fairly computationally intensive and make take a few minutes to run.

(tr.cost1 <- transition(cost1, transitionFunction=mean, directions=8))

We can set our connections based on 4, 8, or 16 neighbor rules. A value of 8 connects all adjacent cells in 8 directions.

class : TransitionLayer
dimensions : 746, 782, 583372 (nrow, ncol, ncell)
resolution : 30, 30 (x, y)
extent : 677292.5, 700752.5, 4987413, 5009793 (xmin, xmax, ymin, ymax)
coord. ref. : +proj=utm +zone=11 +ellps=GRS80 +towgs84=0,0,0,0,0,0,0 +units=m +no_defs
values : conductance
matrix class: dsCMatrix

It is possible using the *transitions* function to have asymmetric transitions. How could we apply asymmetric transitions with genetic data?

Visually inspect your transition raster.

plot(raster(tr.cost1))

Transition calculated transition values based on values of adjacent cells in our "cost" raster. However, we used an 8 neighbor rule and the center of diagonally connected raster cells are farther apart from each other than the orthogonally connected cells. We are using UTM coordinates (i.e., a distance-based projection). However, in lat-long projections cell sizes become shorter as you move N/S toward the poles. Values of the matrix have to be corrected for the first type of distortion for our analysis (and we would need to correct for the second type of distortion if we were using lat/longs).

tr.cost1 <- geoCorrection(tr.cost1,type = "c ", multpl=FALSE)

Step 2 – Create a Cost-Distance Matrix

Now we will create a cost distance matrix based on the above transition conditions. Cost distance matrix is based on the corrected transition layer and the site locations (as a spatial points file). All distance function require conductance values, even though distance will be 1/conductance (friction/resistance).

Our cost distance is a function of the transition (*tr.cost1,* a transition object) and spatial locations (*sites*, a spatial object). This is a single least-cost path between each pair of sites.

cost1.dist <- costDistance(tr.cost1,sites) – **This file is located in the exercise folder**

Step 3 – Create a Cost-Distance based on random paths (analogous to circuit theory)

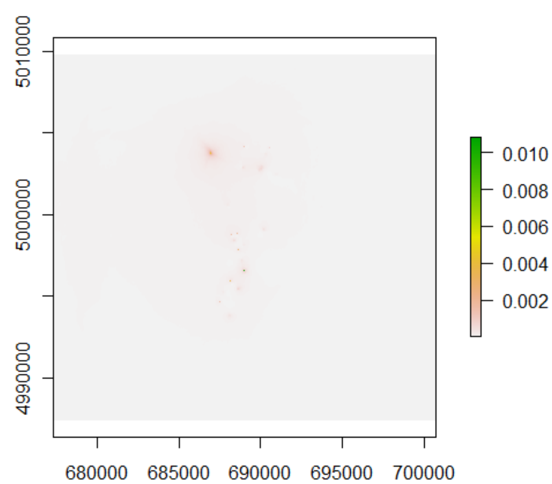
The *passage* function calculated the number of passages through cells before arriving at a destination cell akin to circuitscape (McRae et al. 2008).

Origin points (as a spatial object)
origin <- sites

Destination points (also as a spatial object, in our case origin and destination sites are the same).
goal <- sites

Calculate the number of passages connecting areas. 0 represents a random walk, 1-20 conditions the walk. Again, this process is fairly computationally intensive and may take a few minutes to run.

cost1.cs <- passage (tr.cost1, origin, goal, theta=0)
plot(cost1.cs)



Due to the scale and color ramp, the costs are a little hard to see. **What might you be able to do to improve this plot?**

Part 4: Alternative method for calculating costs

Repeat the above methods, only using the *costnsc.img* raster as your cost raster and name it as niche.cost. It will replace “cost1” in part 2, step 2.

niche.cost<- raster("costnsc.img")

Question. Visually compare the cost surfaces. How different are they?

Question. What are the strengths and weaknesses of using a “best guess” weighting scheme compared to a niche or habitat selection prediction?

Part 5: Correlating cost distance with genetic distance

Step 1 – create geographical distance matrix

Create a matrix of site locations

```
sites.m <- read.csv ("RALU_UTM.csv")
sites.m<-as.matrix(sites.m[c("X", "Y")])
```

Get geographic distance in meters

```
geo.dist <-pointDistance(sites.m,longlat=F)
geo.dist <-as.dist(geo.dist)
```

Step 2 - Import your genetic distance matrix and covert into a distance matrix

```
gen.dist <- read.csv("RALU_DpsFinal.csv")
gen.dist<-as.dist(gen.dist)
```

Question - What genetic distance is being used here? How much do you think the genetic distance might impact results?

Step 3 – Calculate correlations among distance matrices
(geographic distance, genetic distance, alternate estimate of “effective” distance)

Is there isolation by distance in this dataset?

```
cor(gen.dist, geo.dist)
```

Correlation between genetic distance and the first least-cost path calculation

```
cor(gen.dist, cost1.dist)
```

Correlation between genetic distance and total resistance among sites

```
res1.dist <- commuteDistance(tr.cost1,sites)
cor(gen.dist, res1.dist)
```

Also compute the correlations with the niche layer least cost path and niche resistance.

We now have five correlations. **Which one is the “best” model?** We need to control for distance in some way (i.e., is landscape cost a better model than just distance).

We need load the ecodist package for partial mantel tests.

```
mantel (gen.dist ~res1.dist + geo.dist, nperm=1000)
mantel (gen.dist ~cost1.dist + geo.dist, nperm=1000)
```

Try this same procedure with the niche prediction.

How much does resistance estimate matter in this system?

Process	Variable	Code	Source	Pred	Description	Calculation	Ecological Justification
Climate	Growing season precipitation	gsp	Spline	+	Precip	(Rehfeldt 2006)	Moist areas are good for dispersal (Munger *et al.* 1998).
Topo	Elevation relief ratio[1]	err	SRTM	-	Elevational complexity	(Evans 1972)	Fine scale – topographic complexity may make travel energetically expensive. Coarse scale – identify major topography which may be barriers (Funk *et al.* 1999; Lougheed *et al.* 1999; Funk *et al.* 2005a).
Temp-Moist	Compound topographic index	cti	SRTM	+	Flow accumulation by catchment size (wetness)	(Moore et al. 1993)	Species may disperse through wet areas (Pilliod *et al.* 2002; Bartelt & Peterson 2005).
	Frost-free period	ffp	Spline	+	Date of last freeze minus date of first freeze	(Rehfeldt 2006)	Short growing season may result in less dispersal (Palo *et al.* 2003). Ffp is inversely correlated with maximum temperature which impedes dispersal.

References

Evans IS (1972) General geomorphometry, detivatives of altitude, and descriptive statistics. In: Chorley RJ (ed) Spatial Analysis in geomorphology. Harper & Row, New York,

Funk WC, Blouin MS, Corn PS, Maxell BA, Pilliod DS, Amish S, Allendorf FW (2005) Population structure of Columbia spotted frogs (*Rana luteiventris*) is strongly affected by the landscape. Mol Ecol 14 (2):483-496

McRae BH, Dickson BG, Keitt TH, Shah VB (2008) Using circuit theory to model connectivity in ecology, evolution, and conservaiton. Ecology 89 (10):2712-2724

Moore I, Gessler P, Nielsen G, Petersen G (eds) (1993) Terrain attributes and estimation methods and scale effects. Modeling change in environmental systems. Wiley, London

Murphy MA, Dezanni RJ, Pilliod D, Storfer A (2010) Landscape genetics of high mountain frog metapopulations. Mol Ecol 19:3634-3649

Pilliod DS, Peterson CR, Ritson PI (2002) Seasonal migration of Columbia spotted frogs (*Rana luteiventris*) among complementary resources in a high mountain basin. Can J Zool 80:1849–1862

Rehfeldt GE, Crookston NL, Warwell MV, Evans JS (2006) Empirical analysis of plant-climate relationships for western United States. International journal of plant sciences 167:1123-1150