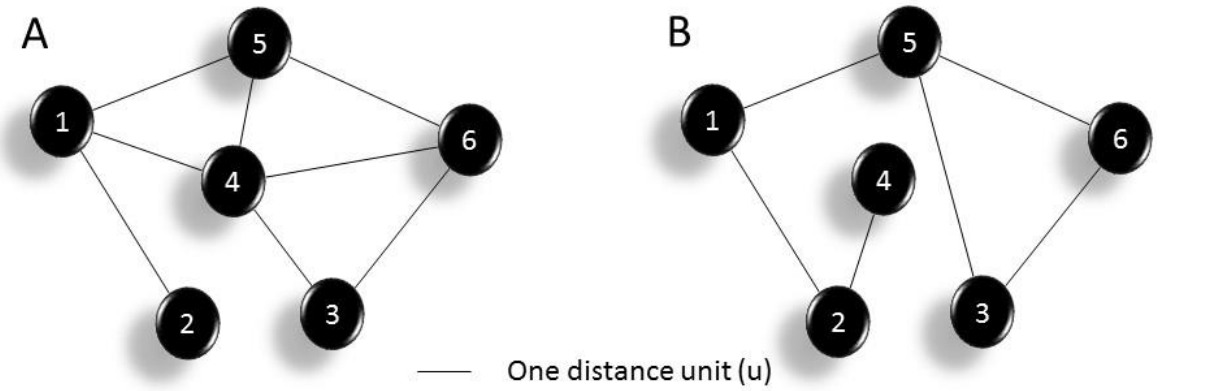
Graphs and Network Models

Melanie Murphy, University of Wyoming

Part 1 – Conceptual Exercise (20 min) – Graph Topology

Graphs A and B represent gene flow among sample locations. In this case, the graphs are relatively simple – links represent presence of gene flow but not amount of gene flow (an edge weight). Even given this relatively simple graphs, we can assess graph topology both at the graph and node levels. Fill out the *Diameter* (length = # of links or you estimate cumulative length of the path) and Degree Distribution below.

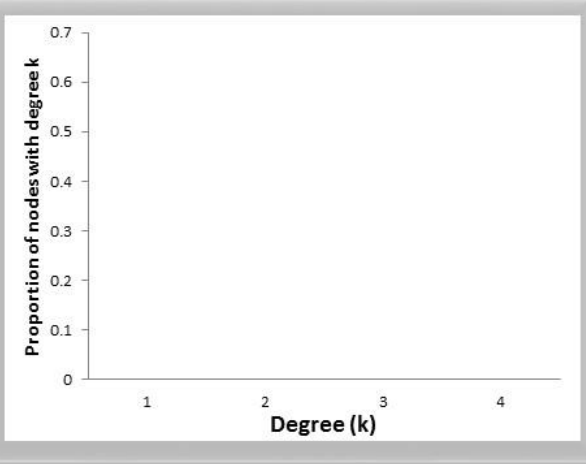


Whole Graph Metrics

	A	B
Diameter		

Diameter is the length of the path separating the two most distant nodes.

Graph Degree distribution for Graph A vs. Graph B . Degree is the number of nodes (k) a given node (n) is connected to in a graph. Proportion of nodes (P(k)=nk/n) is the proportion of nodes with degree k.



Questions:

1.1 Which graph (A or B) has greater diameter? What does greater diameter mean in terms of functional connectivity?

1.2 Describe the degree distribution of Graph A vs. Graph B.

By Node

Node	Degree[1]		Betweeness		Closeness	
	A	B	A	B	A	B
1			4	6	0.14	0.13
2			0	4	0.09	0.10
3			0	0	0.11	0.08
4			3.5	0	0.16	0.07
5			1	6	0.14	0.13
6			.5	0	0.13	0.08

Summarized by Graph

Metric	A	B
Freeman's Degree Centrality	0.4	0.3
Mean Closeness		
Mean Betweeness		

Local graph metrics are intended to quantify the relative importance of nodes, collections of nodes, or edges in the overall structure of the graph topology (Wasserman and Faust, 1994). Useing graphs *A* and *B* fill in (unscaled) degree centrality. I calculated betweeness and closeness for you, but calculate the overall graph summary (mean closeness, mean betweeness).

[1]For ease of calculation, use unscaled degree (number of edges that go through a node). Scaled degree centrality would be # edges /(n-1) where n is the number of nodes.

Questions:

1.3 For each graph, which node(s) has the highest degree? What does this mean?

1.4 For each graph, which node(s) are most important for maintaining graph connectivity? Why?

1.5 Using the summary statistics (table on the right), which graph is more connected?

1.6 Imagine that these graphs are in landscape that are facing rapid habitat lost and fragmentation. Based on the graph metrics, which graph do you think is more vulnerable to breaking into multiple graphs (i.e., losing overall connectivity among samples sites)? Why?

Part 2 – Gravity models (40 min)

For part 2, I recommend that you work through the R code, try to understand what the code is doing and work through the questions. Output are provided, however, so you can work through the exercise without a hands-on R component.

Background: There are many ways graphs can be implemented to understand population structure and relate that structure to landscape characteristics (see Dyer and Nason 2004). In this exercise, we will focus on one specialized case. Gravity models are a type of inferential model that exploit graph characteristics. Gravity models include both at site and between side landscape data. They are a type of graph consisting of nodes and edges. These nodes and edges of landscape characteristics associated with these graph elements. In this exercise, you will use the gravity model framework to build an empirical model of gene flow for the Columbia spotted frog dataset in central Idaho that you have used for several other exercises (Murphy et al. 2010).

References:

Gravity models for landscape genetics: derivation and methods:

Murphy, M. A., R. J. Dezanni, D. Pilliod, A. Storfer (2010). "Landscape genetics of high mountain frog metapopulations." Molecular Ecology **19**: 3634-3649.

Gravity model package:

Murphy, M. A., J.S. Evans. (in prep). "GeNetIT: gravity analysis in R for landscape genetics" Methods in Ecology and Evolution

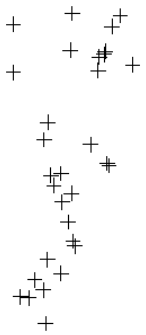
Evans, J.S. , Murphy, MA (2015). *GeNetIT* R package version 0.1-0.

Step 1 – Establish the R Environment

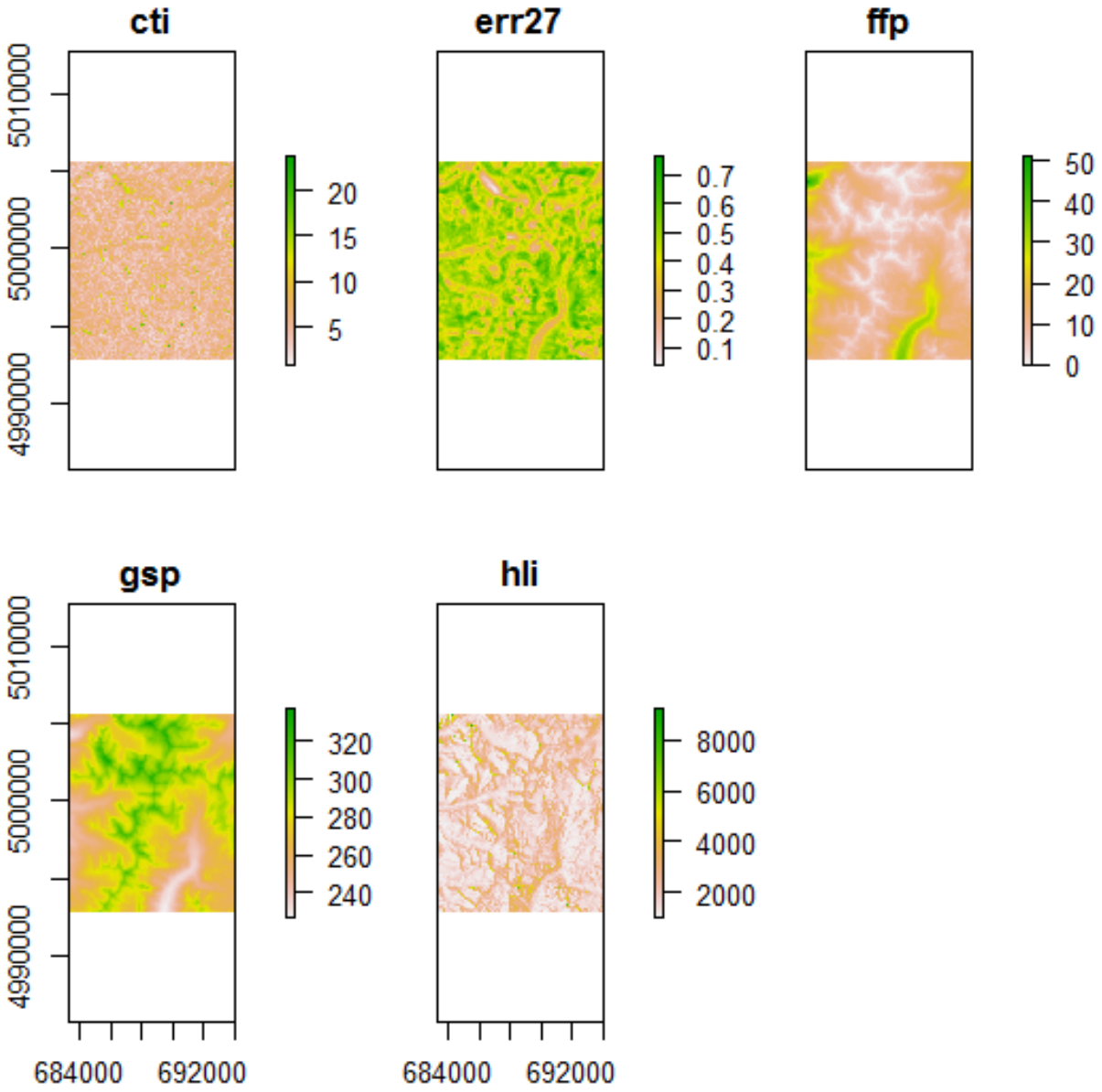
1. Step through the code (gravity.R). Look for errors and stop if an error is reported. Using **WordPad or copying from this document may run into issues due to hidden characters.**
2. If you are running on a mac, you may need to install gdal on your machine. For instructions, see http://www.gis.usu.edu/~chrisg/python/2009/docs/gdal_mac.pdf. Note that these instructions have worked for people in the past, but I don't support Macs for this code.
3. Make sure you have the required packages: **GeNetIt, spatialEco, spdep, maptools, RANN, sp, raster, rgdal**
4. In this exercise, we are going to step through the R code, running one section at a time.
5. Several data files are included with the GeNetIT package.
    a. Dps – genetic distance in proportion of shared alleles.
    b. ralu.site – this is a shape file with wetland data and spatial locations
    c. rasters – 30 m rasters of landscape variables.

```
#install new packages
install.packages("rgdal")
install.packages("spatialEco")
install.packages("GeNetIt")

#require packages (may also need to install some of the below #packages)
require(raster)
require(rgdal)
require(raster)
require(GeNetIt)
require(spdep)
require(maptools)
require(RANN)
require(sp)
require(spatialEco)
require(GeNetIt)

#data for exercise
data(dps)
data(ralu.site)
data(rasters)

# Coerce #SpatialPixelsDataFrame to #raster stack
( xvars <- stack(rasters[-6]) )
( land.cov <- stack(rasters[6]) )

#plot sample locations
plot(ralu.site)
```

.



```
#plot raster data
plot(xvars)
```



Step 2- Build additional at site covariates by extracting raster point values

```
ralu.site@data <- data.frame(ralu.site@data, extract(xvars[[c(1,2)]], ralu.site))
#at each sample site, extracting values from the first and second rasters

#Take a look at the resulting data
head(ralu.site@data)
```

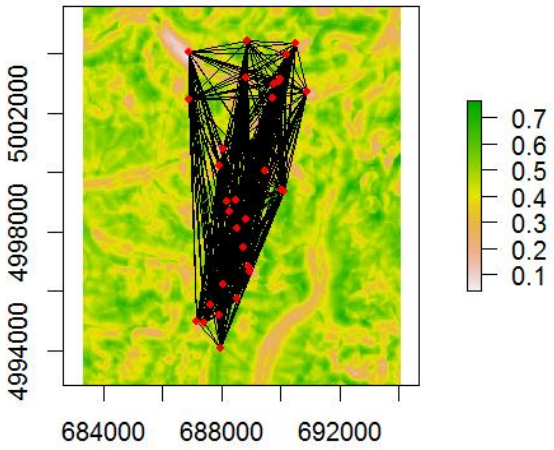
Step 3- Create a kNN graph

The next step is to create a graph. In this case, you will first create a saturated graph (each site connected to all other sites).

```
# 1) Create kNN graph from the site data (ralu.site)
dist.graph <- knn.graph(ralu.site, row.names = ralu.site@data[,"SiteName"])

#you can limit the graph by maximum distance #(max.dist) if desired
#dist.graph <- knn.graph(ralu.site, row.names = ralu.site@data[,"SiteName"], max.dist = 5000)

# 2) Add "from.to" unique ID's and merge with genetic distance matrix
  dist.graph@data$from.to <- paste(dist.graph$i, dist.graph$j, sep=".")
  dps$from.to <- paste(dps$FROM_SITE, dps$TO_SITE, sep=".")
  dist.graph <- merge(dist.graph, dps, by = "from.to")

# 3) Merge graph with at site nodes
  dist.graph@data <- dist.graph@data[,-c(7,8)]

# Remove NA values
  na.index <- unique(as.data.frame(which(is.na(dist.graph@data), arr.ind = TRUE))[,1])
  dist.graph <- dist.graph[-na.index, ]

# Display columns and plot
str(dist.graph@data)
plot(xvars[[2]])
 plot(dist.graph, add=T)
  points(ralu.site, pch=20, col="red")
```

2.1 *What are some other methods for "pruning" this graph?*

What would an ecologically justifiable way of determining a maximum distance?



Step 4- Add covariates to edges by extracting raster values and calculating statistics

```
#can calculate any statistical moment to describe the values between sites
#This examples has min, mean, max and variance
#Example is all for floating point data

stats <- graph.statistics(dist.graph, r = xvars, d=30,
          stats = c("min", "mean", "max", "var"),
          sp = FALSE)
dist.graph@data <- data.frame(dist.graph@data, stats)

##Calculating statistical moments of categorical data does not make sense
#### Example of categorical raster data
####   function for percent wetland landcover (NLCD).

wet.pct <- function(x) {
  x <- ifelse( x == 11 |  x == 12 | x == 90 | x == 95, 1, 0)
#have multiple numeric categories that are wetlands

   prop.table(table(x))[2]
}
lc.stats <- graph.statistics(dist.graph, r = land.cov, d=30,
          stats = "wet.pct")
    lc.stats[is.na(lc.stats)] <- 0
dist.graph@data <- data.frame(dist.graph@data, lc.stats)


str(dist.graph@data)
```

2.2 *What other statistical moments might be informative?*

2.3 *Why are you calculating % of habitat and not amount for nlcd data (wet.pct)?*

2.4 *We know animals (and pollen/seeds) are influenced by more than a 30 m line. How might we account for this reality?*

Step 5 – Gravity model

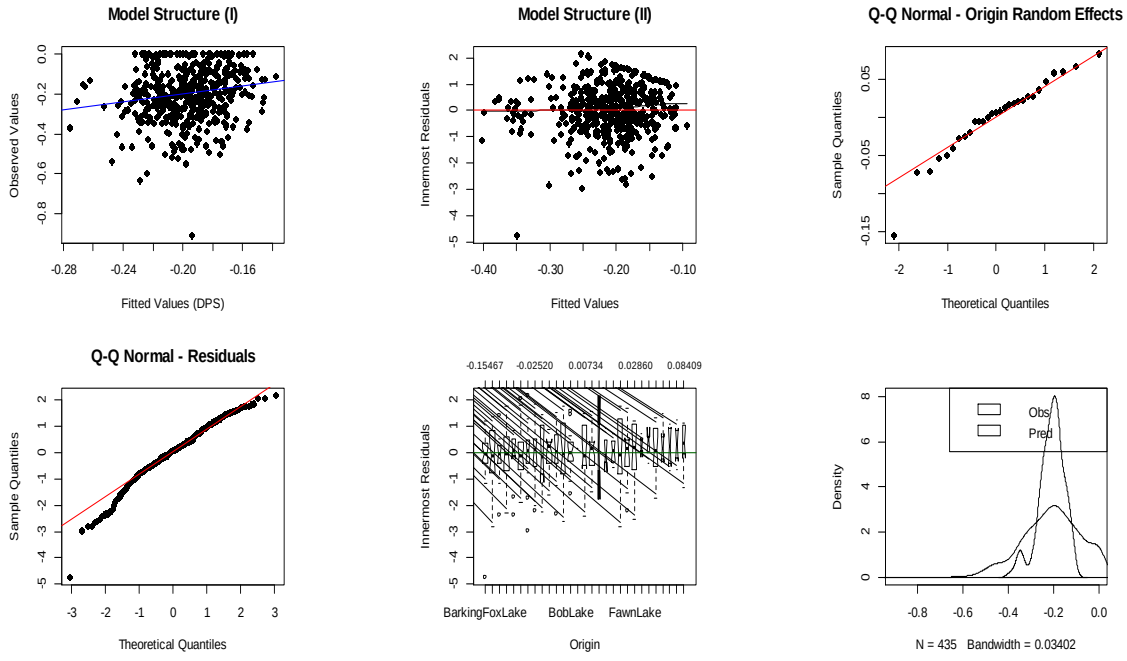
Now we are ready for the actual gravity model analysis. There are a few things to remember:

1. In order to solve the gravity form, we take the natural log of all dependent and independent variables.
2. The dependent variable (y) is the genetic distance (T).
3. The independent variables (x) fall into the three components of the gravity model: distance (w), at site (v), and between site (c).

```
#########################################################################
# Build data for gravity model
#########################################################################

from <- ralu.site@data[,c(1,6,8,18,19)]
  names(from)[2:ncol(from)] <- paste("f", names(from)[2:ncol(from)], sep=".")
to <- ralu.site@data[,c(1,6,8,18,19)]
  names(to)[2:ncol(to)] <- paste("t", names(to)[2:ncol(to)], sep=".")
site <- data.frame(from,to)
  site <- site[,-(dim(to)[2]+1)]

cdata <- merge(dist.graph, site, by.x="from_ID", by.y="SiteName")
cdata$Dps <- 1 - cdata$Dps
cdata <- cdata@data

#########################################################################
# Specify and fit gravity model
#########################################################################
# Specify parameters
#cti – compound topographic index (wetness index)
#err27 – elevation relief ratio (measure of topographic complexity, 27X27 window)
#gsp – growing season precipitation
#f.AREA_m2 – wetland area of the “from” site in meters squared
#f.Depth_m – wetland depth of the “from” site, this is highly correlated with predatory fish
#f.err27 – elevation relief ratio at the wetland
#length is distance between sites in meters

x = c("length", "mean.cti", "mean.err27", "mean.ffp", "mean.gsp",
     "f.AREA_m2", "f.Depth_m", "f.err27")

# Gravity model
#group variable – this sets the constraint.  This is a from node constraint (singly constrained
#production model)

( gm <- gravity(y = "Dps", x = x, d = "length", group = "from_ID", data = cdata) )
# Plot gravity results
par(mfrow=c(2,3))
   for (i in 1:6) { plot(gm, type=i) }
```

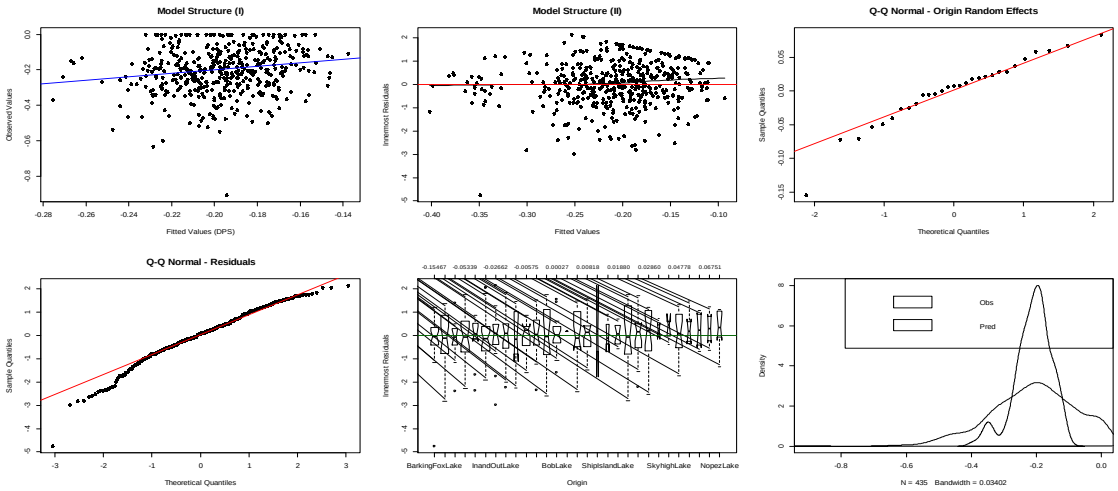
2.5 *The data can be constrained by origin (from site) or destination (to sites). How is the model constrained? How would you change the code to alter the type of constraint? Decide if you think it makes more ecological/biological sense to constrain by origin (From) or destination (To).*

```
[1] "Running singly-constrained gravity model"
Gravity model

Linear mixed-effects model fit by REML
 Data: gdata
        AIC       BIC   logLik
  -526.4635 -481.8647 274.2317

Random effects:
 Formula: ~1 | from_ID
        (Intercept)  Residual
StdDev:  0.06131731 0.1175011

Fixed effects: list(fmla)
                Value Std.Error  DF     t-value p-value
(Intercept) -3.177004  6.597703 401 -0.4815318  0.6304
length      -0.011771  0.007106 401 -1.6565913  0.0984
mean.cti    -0.143304  0.067840 401 -2.1123848  0.0353
mean.err27  -0.097117  0.063505 401 -1.5292830  0.1270
mean.ffp     0.076541  0.100011 401  0.7653228  0.4445
mean.gsp     0.543325  1.133997 401  0.4791236  0.6321
f.AREA_m2   -0.002415  0.008249  25 -0.2927442  0.7721
f.Depth_m   -0.004874  0.018645  25 -0.2613980  0.7959
f.err27     -0.036431  0.033103  25 -1.1005267  0.2816
 Correlation:
            (Intr) length men.ct mn.r27 mn.ffp mn.gsp f.AREA f.Dpt_
length       0.062
mean.cti     0.020  0.060
mean.err27 -0.078  0.061  0.357
mean.ffp   -0.970 -0.059 -0.091  0.037
mean.gsp   -1.000 -0.072 -0.034  0.080  0.969
f.AREA_m2  -0.053  0.024  0.051  0.052  0.028  0.046
f.Depth_m   0.018 -0.015 -0.059 -0.062  0.021 -0.015 -0.707
f.err27    -0.077  0.020 -0.003  0.034  0.068  0.080  0.300 -0.068

Standardized Within-Group Residuals:
        Min          Q1         Med          Q3         Max
-4.75221031 -0.53231103  0.07447355  0.62064827  2.15428090

Number of Observations: 435
Number of Groups: 29
```



```
# Try a second model.
x = c("length", "mean.cti", "mean.err27", "mean.ffp", “wet.pct.nlcd”,   "f.Depth_m", "f.cti")
( gm.2 <- gravity(y = "Dps", x = x, d = "length", group = "from_ID", data = cdata) )
par(mfrow=c(2,3))
  for (i in 1:6) { plot(gm, type=i) }
```

```
[1] "Running singly-constrained gravity model"
Gravity model

Linear mixed-effects model fit by REML
 Data: gdata
       AIC       BIC   logLik
  -526.4635 -481.8647 274.2317

Random effects:
 Formula: ~1 | from_ID
        (Intercept)  Residual
StdDev:  0.06131731 0.1175011

Fixed effects: list(fmla)
               Value Std.Error  DF    t-value p-value
(Intercept) -3.177004  6.597703 401 -0.4815318  0.6304
length      -0.011771  0.007106 401 -1.6565913  0.0984
mean.cti    -0.143304  0.067840 401 -2.1123848  0.0353
mean.err27  -0.097117  0.063505 401 -1.5292830  0.1270
mean.ffp     0.076541  0.100011 401  0.7653228  0.4445
mean.gsp     0.543325  1.133997 401  0.4791236  0.6321
f.AREA_m2   -0.002415  0.008249  25 -0.2927442  0.7721
f.Depth_m   -0.004874  0.018645  25 -0.2613980  0.7959
f.err27     -0.036431  0.033103  25 -1.1005267  0.2816
 Correlation:
            (Intr) length men.ct mn.r27 mn.ffp mn.gsp f.AREA f.Dpt_
length       0.062
mean.cti     0.020  0.060                                          |
mean.err27 -0.078  0.061  0.357
mean.ffp   -0.970 -0.059 -0.091  0.037
mean.gsp   -1.000 -0.072 -0.034  0.080  0.969
f.AREA_m2  -0.053  0.024  0.051  0.052  0.028  0.046
f.Depth_m   0.018 -0.015 -0.059 -0.062  0.021 -0.015 -0.707
f.err27    -0.077  0.020 -0.003  0.034  0.068  0.080  0.300 -0.068

Standardized Within-Group Residuals:
        Min          Q1         Med          Q3         Max
-4.75221031 -0.53231103  0.07447355  0.62064827  2.15428090

Number of Observations: 435
Number of Groups: 29
```



Part 3 – Building your own models and model selection (30 min)

Now is your opportunity to build your own models. Apply what you learned from the Model Selection lab. Select 5 candidate models to test with a clear ecological/biological *a priori* justification – all of these models should include the distance (length) term and the grouping factor. Please include both a null model (grouping factor, distance, no other parameters) and a global model. Use the correlation matrixes to test for collinearity among model terms. You will use your information to fill out the table below. To see potential variables, type names(cdata).

Description of potential variables:

length = distance between sites (in meters)
cti = compound topographic index
hli = heat load index
err = elevation relief ratio (27X27 window)
ffp = frost free period
gsp = growing season precipitation
wet = percent wetland (CHALLENGE- you could calculate other habitat variables from the nlcd data)
.var = variance (of values along the edge)
.min = minimum value (of values along the edge)
.max = maximum value (of values along the edge)
.mean = mean value (of values along the edge)
f.[variablename] = from node variable
t.[variablename] = to node variable

```
# Modify this code  for your models
x = c("length", #insert names of your variables)

( gm.X <- gravity(y = "Dps", x = x, d = "length", group = "from_ID", data = cdata) )
#Could change the grouping factor, but use either from or to across all models

#plot the result
par(mfrow=c(2,3))
  for (i in 1:6) { plot(gm, type=i) }
```

For each model, contemplate the diagnostic plots. Think about idealized expectations for each plot and how your models/data may diverge from these expectations. Below are the different plots.

- ✓ Plot 1 – Observed vs. expected.
- ✓ Plot 2 – residual error
- ✓ Plot 3 - Q-Q Normal - Origin
- ✓ Plot 4 – Q-Q Normal – residuals
- ✓ Plot 5 – Fitted values by origin (or destination, this is by the "grouped" factor).
- ✓ Plot 6 – Density, observed vs. expected.

3.1 *What is the meaning of plot 5?*

Model	AIC	BIC	Notes from plots
1.			
2.			
3.			
4.			
5.			

3.2 *Is there any evidence that you may be violating regression assumptions for any of the models? What steps could you take to accommodate any observed (potential) assumption violations?*

3.3 *Which model(s) have the highest support? Do you get the same results with AIC compared to BIC scores?*

3.4 *What additional variables might you hypothesize are related to production/attraction and/or resistance among sites?*