MONTE CARLO ANALYSIS: ESTIMATING GPP WITH THE CANOPY CONDUCTANCE METHOD

1. Overview

A novel method to measure Gross Primary Production (GPP) of forested ecosystems was recently developed based on sap-flow derived estimates of canopy conductance and stable-isotope derived measurements of c_i/c_a. We have proposed to use this method to investigate age-related variance in GPP across a chronosequence of loblolly pine forests ranging from 14- to 115-years old. As this method has not been widely employed, we performed a Monte Carlo Analysis to investigate the power of our statistical approach: i.e. what is the smallest change in GPP we can expect to detect given the natural variance in GPP?

2. Approach and Assumptions

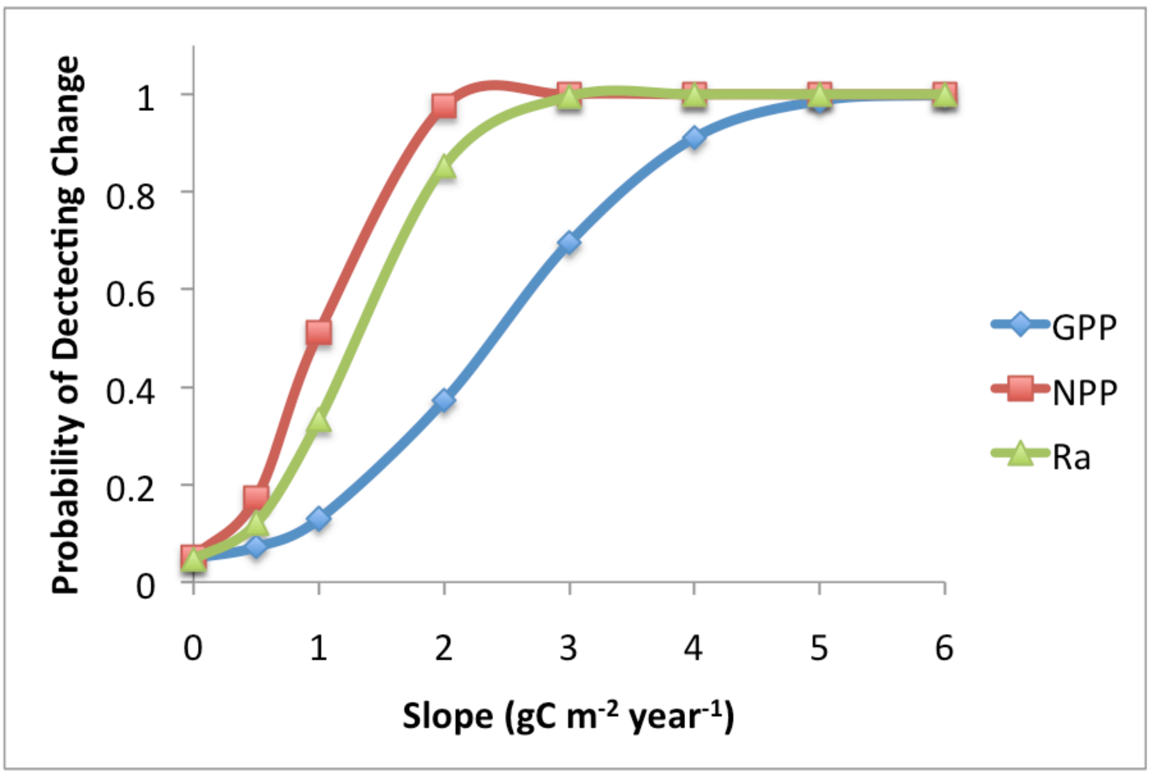
The Monte Carlo Analysis was performed as follows:

- **Natural variation**. The only study to date using this approach estimated GPP in a 20-year old loblolly pine forest to be 2287 gC m^{-2} $year^{-1}$ with a standard deviation of 148. We assumed that this standard deviation was constant across similar forests of different age. We also ran a Monte Carlo analysis for Net Primary Production (NPP) assuming a mean of 909 gC m^{-2} $year^{-1}$ with a standard deviation of 54 and Autotrophic Respiration (Ra) assuming a mean of 1014 gC m^{-2} $year^{-1}$ with a standard deviation of 71. All values are from Schafer *et al.* (2003), Global Change Biology.
- **Forest Age**. Nine forests of known age have already been selected, so the data for the x-axis are known. The ages are: 14, 19, 22, 36, 42, 53, 97, 114, and 115 years old.
- **Power Analysis**. Most packages for regression power analyses do not allow for situations where the x-axis data are known. Therefore we wrote our own program in the R environment. The program is presented as an appendix to this document.
- **User Inputs**. The user inputs the "true" slope to evaluate, and the program generates simulated data for each of the nine forests according to a normal distribution derived from the standard deviation. The program then does a regression analysis of these data and writes the p-value to a file. This is

Date: June 27, 2008.

repeated 10,000 times, resulting in 10,000 p-values of possible observations given a "true" change in GPP. The fraction of p-values less than 0.05 are presented in the figure on the next page.

3. Results



The Monte Carlo analysis indicated that measurements of GPP using the Canopy Conductance Method in the nine identified forest stands would have >95% chance of detecting a decline in GPP that is greater or equal to 5 gC m^{-2} $year^{-1}$. This corresponds to a change in GPP from 2461 gC m^{-2} $year^{-1}$ to 1956 gC m^{-2} $year^{-1}$: a decline of 20%. While we have the capacity to detect smaller changes in Ra and NPP, an ecologically meaningful change in GPP is likely to be detected. We conclude that the canopy conductance approach is a reasonable method with which to investigate age-related change in GPP.

4. Appendix- Function Code

```
sense.slope <- function(mean,std,slope,reps,outputpath) {
    count.05 <- 0                          # This will be used as a counter for sig. p-values, so it is set to zero at the beginning.
    setwd("/Users/John/R")
    age <- c(14,19,22,36,42,53,97,114,115) # my actual observed ages
    intercept <- (mean + 20 * slope * -1)       # estimates intercept anchored to Schafer (2003)'s observation
    meanGPP <- age * slope + intercept

# This loop creates as many p-values as reps you have asked for

        for (n in 1:reps){
            estGPP <- c(1,1,1,1,1,1,1,1,1)      #creates a dummy vector of the right length
            for (i in 1:length(meanGPP)){
                estGPP[[i]] <- rnorm(1,mean=meanGPP[[i]],sd=std)
            }#takes ONE random pull of a GPP values for each site age

            matrix <- cbind(age,estGPP)
            linearmodel <- lm(age~estGPP)
            a <- summary(linearmodel)
            f.stat <- a$fstatistic
            p.value <- 1-pf(f.stat["value"],f.stat["numdf"],f.stat["dendf"])
            write(p.value,file=outputpath,append=TRUE)
            if (p.value < 0.05) {count.05 <- count.05 + 1}
        }

    results <- read.table(file=outputpath)
    hist(results$V1,probability=TRUE)
    #write.csv(matrix,file="matrix.csv")
    proportionsig <- count.05 / reps
    proportionsig # returns the proportion of p-values that were significant

}
```