

## PROJECT 1

---

This is a small project that deals with plotting and doing general calculations in R by examining the pharmacokinetics of a drug, indomethicin.

Tasks:

1. Load the “**Indometh**” dataset and look it over to find out what’s in it.  
*Functions: data*
2. Plot curves of concentration vs. time for each subject, all in a single plot.  
*Functions: plot, lines*
3. Calculate the rate of decay for each person and the global average.
4. Calculate the half-life for each sample and the global average.

## PROJECT 2

---

This is a small project examining the survival of a group of patients with melanoma.

Tasks:

1. Load the “**Melanoma**” dataset, which is in the “**MASS**” library. Examine the data to find out what’s in it.  
*Functions: library, data*
2. Determine if there are obvious differences in the mean or standard deviation of the various outcomes.  
*Functions: mean, sd, tapply*
3. Use a chi-square test to determine if ulcers have an effect on outcome. Is there any difference if the p-value is simulated or just calculated?  
*Functions: chisq.test*
4. Is there any relationship between age and outcome? Age and thickness?
5. Is there a relationship between survival time and thickness?
6. Is there a relationship between sex and survival time?

## PROJECT 3

---

This is a “make your own project” project.

Tasks:

1. Using the built-in Data Manager in R, find a dataset that interests you. Load it into R.  
*Commands: data*
2. Examine and analyze the data as you see fit.

## PROJECT 4

---

This is a complex project which will probably keep you busy for a while. We're going to use BioConductor to analyze Affymetrix GeneChip data from an experiment on prostate cancer<sup>1</sup>. You can read about the data at <http://scgap.systemsbiology.net/data/>.

Tasks:

1. Install BioConductor and the relevant packages (ask me how).
2. Copy over the data files to your hard drive (ask me how).
3. Read over the “BioConductor Cheat Sheet” which has thoughtfully been prepared for you.
4. Load the Affymetrix CEL files into R. Use the “phenodata.txt” file for your phenotypic data.  
*Functions: ReadAffy*
5. Perform data “cleaning” and summarization to derive gene expression profiles. Some algorithms are very slow, so in the interest of time, use the following methods: “rma” for background correction, “quantiles” for normalization, “pmonly” for PM adjustment, and “avgdiff” for summarization.  
*Functions: espresso*
6. Analyze the expression data to find the top 50 differentially expressed genes. Obtain English-language descriptions of those genes. Hint: the “**lmFit**” function uses a parameter called “design” which describes how the data is structured. Just pass in the following:  
`cbind(WT=c(1,1,0,0), MU=c(0,0,1,1))`  
I'll be glad to explain what it means if you're interested.  
*Functions: lmFit, eBayes, topTable, aafDescription*
7. Produce a heatmap of the top-50 with dendrograms on the top and side. Hint: you'll need the “**which**” function to get the data for the top-50 items.  
*Functions: which, heatmap*
8. Analyze the top-50 for overrepresented GO categories. Hint: the “**GOHyperG**” function requires a list of unique LocusLink IDs. Use the “**unique**” function to remove duplicates from a vector.  
*Functions: unique, as.numeric, aafLocusLink, GOHyperG, names*

---

<sup>1</sup> Oudes AJ, Campbell DS, Sorensen CM, Walashek LS, True LD, Liu AY. Transcriptomes of human prostate cells. BMC Genomics. 2006 Apr 25;7:92.