Stat 430 Homework 1

due: October 1, 2009

This (and future) homeworks will require you have access to R. You can install R on your own computer, but R is also availabe on the thin clients in the various computer labs of Snedecor hall. As a student you can also connect to the Statistics Department's terminal server. I will indicate how to do this in Windows. Open Remote Desktop Connection (Start → Program Files → Accessories) and open a connection to `ts.stat.iastate.edu` with your ISU username/password. This is the same server the thin clients use.

When turning in the homework, please email code that *runs*. The answers should either be printed out by the code (see function `pdf()` for writing pdf graphics files) or you must also submit another *single* document (paper copy or forwarded in email) that neatly displays *all* the answers obtained from the code. In all, you should send or deliver to me *at most two files* for this homework. Please comment your code and use `print()` statements to explain any output your code produces.

1. The MASS package contains a dataset `Cars93`, which you can load with the command `data(Cars93)`. This dataset lists the types of cars on sale in the USA in 1993.

   (a) There are several types or categories of cars, listed under the `Type` column. Find the cheapest car in each category.

   (b) Compute the mean horsepower and mean city mileage (MPG) for each type.

   (c) What other quantitative variable in this dataset, other than highway mileage, shows the highest absolute correlation with city mileage? Report the variable and the maximum correlation.

   (d) Use `write.table()` to save the manufacturer, model, type, price and mileage data for cars of US origin to a file. Load the data from the file and report the output of the `head()` command applied to the loaded dataset.

   (e) Plot a histogram of city mileage. Does it look normally distributed?

   (f) Use the command `sample()` to sample $n = 10$ cars and compute the mean city mileage. Now, repeat $B = 999$ times, collecting all $1,000$ means, and plot a histogram of sample means. Does the result normal? If not, does it look normal if $n = 30$ and $B = 1000$?

2. Design and implement a simulation study to confirm the claim that if random variable $X \sim$Poisson(λ) represents the number of events, and each event is independently marked with probability p, then Y, the number of marked events follows a Poisson(λp) distribution.

   (We have now learned how to formally test whether the simulation data agrees with a Poisson(λp) distribution. I am not asking for a formal goodness-of-fit. Visual confirmation of a match is sufficient.)

3. The activity of web servers can be measured by counting incoming http requests. For a certain server, the counts in consecutive five-minute intervals may be regarded (approximately) as repeated independent observations from a normal distribution. Suppose the mean five-minute count for this server is $1,200$ and the standard deviation is 35. For that server, let Y represent a five-minute count and let $\bar{Y}$ represent the mean of size five-minute counts. Find $P(1175 \leq Y \leq 1225)$ and $P(1175 \leq \bar{Y} \leq 1225)$ and compare the two. Does the comparison indicate that counting for thirty minutes and dividing by six would tend to give a more precise result than merely counting for a single five-minute interval? How?

4. It is well known that one in ten users of a certain type of software call technical support for assistance. If a company sells 15 copies of the software and receives no calls, what would be their estimate of the proportion of *their clients* who call technical support? Construct a 95% confidence interval for this estimate and justify your method of construction. What is wrong with the substitution method for this application?

5. Derive the maximum likelihood estimates of λ and p if you observe n independent pairs (X_i, Y_i) (note X_i is not independent of Y_i), where $X_i \sim$Poisson(λ) and Y_i is the number of marked (independently with probability p) events in X_i. (See problem 2.)

6. [added] The ABO blood locus is a gene on chromosome 9 where one of three gene variants (called alleles) are possible: A, B, or O. Humans have two copies of every gene, so they have two ABO alleles. Individuals who have two A's, denoted by genotype AA, are blood type A, but so are individuals with genotype AO. The correspondence between blood types and genotypes at the ABO locus are shown in the table.

Blood Type	Type A	Type B	Type O	Type AB
Genotypes	AA or AO	BB or BO	OO	AB
Probability	$p_A^2 + 2p_A p_O$	$p_B^2 + 2p_B p_O$	p_O^2	$2p_A p_B$
Armenia	0.31	0.50	0.13	0.06
Data	52	37	11	0

The third row of the table proposes a model for the probability of each category based on the population probability of each allele: p_A, p_B, and p_O. The fourth row presents the known genotype proportions in the Armenian population. The last row presents some data on the number of individuals in each category from a sample of size $n = 100$

(a) Use the data to compute the p-value for testing the hypothesis $H_O : P_A = 0.31$, where P_A is the probability of type A individuals.

(b) Perform a goodness-of-fit test to determine if the observed data are consistent with the Armenian population proportions. Compare the results with the above test. Which test, do you think, has more *power* to reject H_0 when it is not true. I'm not asking for a quantitative answer: guess and justify your choice with words.

(c) In the US, the allele probabilities are known to be $p_A = 0.21, p_B = 0.06$, and $p_O = 0.73$. Use the probability model to test whether the data are consistent with the US population.

(d) Use `R` function `optim` to numerically find the maximum likelihood estimates $\hat{p}_A$ $\hat{p}_B$ for this data set. Notice $\hat{p}_O = 1 - \hat{p}_A - \hat{p}_B$ need not be estimated. Also, the parameters are constrained such that $0 \leq p_A + p_B \leq 1$. It is difficult to impose this constraint directly via `optim`, so I suggest adding the following code to `optim()`'s required `fn` function that you must write:

```
# suppose p is the vector of parameters (p_A,p_B)
# the following code keeps p properly constrained
if(sum(p)>1) p <- p/(sum(p)+2e6)
p[1] <- max(p[1], 1e-6)
p[2] <- max(p[2], 1e-6)
p[1] <- min(p[1], 1-1e-6)
p[2] <- min(p[2], 1-1e-6)
```

For additional help on how to use `optim()`, please see the Examples at the bottom of the `optim()` help file (`?optim`).