A few quotes

“... if we aim, in science, at a high information content ... then we have to admit that we also aim at a low probability.”

“...only...an improbable theory is worth testing.”

- Karl Popper

“We didn’t collect enough data in our experiment, so we had to use statistics.”

- student lab report (apocryphal)
The Challenge for Discovery

Take one or several analyses which yield multivariate spectra of observed quantities, with multiple imperfectly understood background sources, and combine them together to obtain a quantitative measure of the level of significance for the presence of some (possibly unknown) new signal process, or, in the absence of a signal, determine quantitative exclusion bounds on the possibility of its existence, in one or more dimensions, assuming various ranges and values of theoretical model parameters.
Course Goals

- Learn probability and statistics, including
  - probability distributions
  - Bayes/frequentist paradigms
  - parameter estimation
  - confidence intervals
  - systematic uncertainties
  - multivariate techniques
- You will learn to do your own calculations of these things, rather than running canned software!
Course Format

• two lectures per week

• plenty of time for discussion and practical examples

• three homework sets with mathematical and practical (write code) problems

• final project on topic of your choice with short presentations and very short paper
  • presentations Mar 16
  • paper due Mar 18
Course Texts

- Statistical Data Analysis - G. Cowan (Oxford)
- Review of Particle Properties (pdg.lbl.gov) writeups on probability and statistics
- papers, etc. that I will attach to the course webpage
- working on learning SmartSite
- lectures will be posted on line but a great deal will be on the blackboard - may want to take notes
 Software

• most high energy physicists today use ROOT for statistical data analysis, plotting, etc.

• ROOT was written in C++ by the same authors of the earlier dominant Fortran package, PAW

• problem: I am a ROOT novice myself

• the intent is for you to perform calculations yourself, using programs you write:
  • CINT interpreter in ROOT
  • raw C or C++
  • Fortran (77 or 90)
Software

- Mathematica - I won’t object!
- MathCad - ditto
- R - high-level language for statistics calculations
- GNUPLOT - publicly available plotting package
- TopDrawer - theorists still use it!
- NTU - my own Fortran-based ntuple/histogram/graphics package (inc. NTV program)
Computers

- There is an advantage to putting software on your own laptop - do it!
- You need to learn basic Unix/C at a minimum
- Work with each other to get set up
Probability

• we live in a random world...
  • what does random mean?

• particle/nuclear physics:
  • every collision event is different
  • every event is independent

• definition of probability?

THE FREQUENTIST/BAYES WARS!
Frequentist probability

- probability is a frequency for some outcome to occur from a certain ensemble of possible outcomes
- frequentist probability is often expressed (or thought of) as the limit of a ratio, for N trials, and outcome i:
  \[ p_i = \lim_{N \to \infty} \frac{n_i}{N} \]
- ensemble characterized by vector of parameters \( \alpha \)
- outcome is a vector \( x \); frequentist probabilities are of the form
  \[ \mathcal{P}(\bar{x} | \bar{\alpha}) \]
The Ensemble

- the space of all possible outcomes for a frequentist probability is the **ensemble**

- discrete ensemble: elements of a set; may be multiple copies of identical elements

- continuous ensemble: bounded or unbounded range

- thought examples:
  - sum of faces of fair dice (one or several)
  - measurement of a mass
  - number of cone 0.5 $k_T$ jets in a pp collision event
Conditional Probability

- ensemble may be labeled with a vector (set of characteristics)
- subsets of ensemble have specific values for certain vector elements

\[ P(A|B) = \frac{P(A \cap B)}{P(B)} \]
Bayesian probability

- Rev. Bayes: attempting to prove the existence of a supreme being, using logic and mathematics

- we call it Bayesian because much of Bayesian statistics utilizes Bayes’ Theorem:

\[ P(A|B) = \frac{P(B|A)P(A)}{P(B)} \]

- in our terms, Bayesians only talk about probabilities of the form

\[ P(\bar{\alpha}|\bar{x}) \]
Priors

\[ P(A|B) = \frac{P(B|A)P(A)}{P(B)} \]

- in this expression, the value of \( P(A) \) is referred to as the prior probability for \( A \)
- priors can come from
  - prior knowledge
  - other experiments
  - our prejudices
Three Thought Problems

You have a test for AIDS which is “99%” accurate meaning that if you have AIDS, the test says “yes” 99% of the time, and if you don’t it says no 99% of the time. If 1% of the population has AIDS, and you take the test, and it says “yes”, what is the probability you have AIDS?

You have a test beam made from protons on a beryllium target. Your particle ID detector tells you whether you have pions or protons with 90% accuracy...it tells you that you have a pion. What is the probability this is true?

Your analysis indicates that you have a $3\sigma$ excess consistent with the Higgs boson decaying to taus. What is the probability that it is the Higgs?
Monte Hall Problem

- you have three doors
- the prize is behind one of them
- you choose door 1
- Monte Hall opens door 3 - no prize!
- Monte says you can stick or switch to door 2
- is it better to stick with door 1 or switch to door 2, or does it not matter?

Answer Wednesday!
Bayes vs. Frequentism

- The “war” is an old one, and both sides are entrenched.
- We are all Bayesians some of the time, and frequentists some of the time.
- Bayesian probability: scientific answer
  Frequentist probability: scientific question
- “Bayesians address the question everyone is interested in, by using assumptions no one believes, and frequentists use impeccable logic to deal with an issue of no interest to anyone”

  - L. Lyons, SCMA IV, June 2006
Combinatorics

- in particle physics we are very often combining things: hits into tracks, jets into particles, etc.
- often we encounter multiple combinations meeting our needs...must be systematic!
- code example: combine all track pairs in list

```cpp
for(i=1;i<ntrack,i++){
    for(j=i+1,ntrack){
        combineTrack(i,j);
    }
}
```
Combinatorics

- often times algorithms have a lot of these combination loops and it is critical to reduce them for program execution speed

- we will see many examples of this later...

- simple rules:
  - keep (ordered) lists internally
  - calculate whatever you can on the fly
  - ignore stupid combinations
Binomial Coefficient

• if we have \( n \) objects and we combine \( k \) of them a time, the number of unordered combinations is given by the binomial coefficient

\[ nC_k = \binom{n}{k} = \frac{n!}{k!(n-k)!} \]

• \( n \) ways of picking the first object, times \( n-1 \) possibilities for the second, ... \( n-k \) for the last; then there are \( k! \) permutations of the \( k \) objects

• these numbers can get big in a hurry...do you know any calculation tricks?
Random Numbers

- if they come from a computer program, they aren’t random...“pseudorandom” number generators
- many generators are flawed - caveat emptor!
- common problem: hyperplanes in nD space
- all of them repeat with some frequency
- sequences initiated with “seeds”
- generally output in the range (0,1)
- can get anything we need from this!
“Good” Random Generators

- RANMAR - Fortran (CERNLIB); F. James (CERN), pairs of seeds allow parallel usage FLAWED
- RANF - Fortran built-in FLAWED
- ROOT: TRandom? See J. Heinrich CDF note:

4 Recommendations

- Do not use ROOT class TRandom.
- Use the ROOT class TRandom3 instead. TRandom2 is also usable, but not quite as good as TRandom3.
- Do not use ROOT’s TRandom::Poisson, TRandom::PoissonD, TRandom2::Poisson, TRandom2::PoissonD, TRandom3::Poisson, TRandom3::PoissonD.
- Do not use CERNLIB’s Poisson generator RNPSSN.
Distributions

- We will focus the next lecture on distributions, including
  - standard probability distributions
  - general distributions
  - properties of distributions
  - probability densities
  - generating random deviates from general distributions: the core of MC methods