Statistical tests (part I)

1. Hypotheses, test statistics, significance level, power
2. An example with particle selection
3. The Neyman-Pearson lemma
4. Constructing a test statistic:
   
   Fisher discriminant function
   
   Neural networks
Hypotheses, test statistics

Suppose the result of a measurement is \( \vec{x} = (x_1, \ldots, x_n) \)
e.g. events from \( e^+ e^- \) collisions; for each event measure
\[
\begin{align*}
x_1 &= \text{number of charged particles produced} \\
x_2 &= \text{mean } p_\perp \text{ of particles} \\
x_3 &= \text{number of ‘jets’ (according to some algorithm)} \\
x_4 &= \ldots
\end{align*}
\]
\( \vec{x} \) follows some joint pdf in an \( n \)-dimensional space, which depends
on the type of event produced, i.e. \( e^+ e^- \to q\bar{q}, e^+ e^- \to WW, \) etc.
That is, the joint pdf \( f(\vec{x}) \) is specified by a certain

HYPOTHESIS

i.e. predicted probability densities \( f(\vec{x}|H_0), f(\vec{x}|H_1), \) etc.
(Note sloppy but traditional notation: usually \( H_0, H_1, \ldots \) not r.v.s.)

Simple hypothesis: \( f(\vec{x}) \) completely specified,
Composite hypothesis: form of \( f(\vec{x}; \theta) \) given, parameter \( \theta \) unknown.

Usually awkward to work with multidimensional \( \vec{x} \),
\[ \Rightarrow \text{construct test statistic of lower dimension (e.g. scalar), } t(\vec{x}): \]
compactify data,
try not to lose ability to discriminate between hypotheses.

The statistic \( t \) then has pdfs \( g(t|H_0), g(t|H_1), \ldots \)
Critical region, errors of 1st and 2nd kind

Consider a test statistic $t$ following $g(t|H_0)$, $g(t|H_1)$, …

Define a critical region where $t$ is not likely to occur if $H_0$ is true,

e.g. for the case above, $t \geq t_{\text{cut}}$.

If observed value $t_{\text{obs}}$ is in critical region, reject $H_0$, otherwise ‘accept’.

Probability to reject $H_0$ if it is true (error of 1st kind):

$$\alpha = \int_{t_{\text{cut}}}^{\infty} g(t|H_0) \, dt \quad \text{(significance level)}$$

Probability to accept $H_0$ if $H_1$ is true (error of 2nd kind):

$$\beta = \int_{-\infty}^{t_{\text{cut}}} g(t|H_1) \, dt \quad (1 - \beta = \text{power})$$
An example with particle selection

Suppose we obtain \( n \) energy loss measurements for a particle in a drift chamber, construct \( t = \) truncated mean of the measurements, and suppose we know the particles are either electrons or pions:

\[
H_0 = \text{electron (signal)} \\
H_1 = \text{pion (background)}
\]

Select electrons by requiring \( t < t_{\text{cut}} \). The selection efficiencies are:

\[
\varepsilon_e = \int_{-\infty}^{t_{\text{cut}}} g(t|e) \, dt = 1 - \alpha
\]

\[
\varepsilon_\pi = \int_{-\infty}^{t_{\text{cut}}} g(t|\pi) \, dt = \beta
\]

Loose cut: most e accepted, lots of \( \pi \) background
Tight cut: low signal efficiency, pure sample

Fractions of e, \( \pi \) may be unknown; \( t \) follows

\[
f(t; a_e) = a_e g(t|e) + (1 - a_e) g(t|\pi)
\]

\[\rightarrow\] estimate \( a_e \) (for now assume \( a_e, a_\pi = 1 - a_e \) known)
Purity of selected sample

For a measured value $t$, what is the probability to be $e/\pi$?

$$h(e|t) = \frac{a_e g(t|e)}{a_e g(t|e) + a_\pi g(t|\pi)}$$

(Bayes’ theorem)

$$h(\pi|t) = \frac{a_\pi g(t|\pi)}{a_e g(t|e) + a_\pi g(t|\pi)}$$

Bayesian: degree of belief that this particle is $e$ or $\pi$

Frequentist: fraction of particles at given $t$ which are $e/\pi$

→ here both approaches make sense

Often want purity of selected sample:

$$p_e = \frac{\text{number of electrons with } t < t_{\text{cut}}}{\text{number of all particles with } t < t_{\text{cut}}}$$

$$= \frac{\int_{-\infty}^{t_{\text{cut}}} a_e g(t|e) \, dt}{\int_{-\infty}^{t_{\text{cut}}} (a_e g(t|e) + (1 - a_e) g(t|\pi)) \, dt}$$

$$= \frac{\int_{-\infty}^{t_{\text{cut}}} h(e|t) \, f(t) \, dt}{\int_{-\infty}^{t_{\text{cut}}} f(t) \, dt}$$

= electron probability averaged over interval $(-\infty, t_{\text{cut}}]$

Sometimes $h(e|t)$ is reinterpreted as the test statistic;

in principle OK, but but you need to know electron fraction $a_e$. 

---

G. Cowan – Computing and Statistical Data Analysis – University of London Postgraduate Lectures
The Neyman–Pearson lemma

Consider a multidimensional test statistic \( \vec{t} = (t_1, \ldots, t_m) \); hypotheses \( H_0 \) (‘signal’) and \( H_1 \) (‘background’).

What is the optimal choice of the critical region (i.e. cuts)?

The Neyman–Pearson lemma states: to get the highest purity for a given efficiency, (i.e. highest power for a given significance level), choose the acceptance region such that

\[
g(t|H_0) > c,
\]

where \( c \) = constant which determines the efficiency.

(For a proof see Brandt Chapter 8.) Value of \( c \) left open; choose this depending on what efficiency you want.

Equivalently, the optimal scalar test statistic is

\[
r = \frac{g(t|H_0)}{g(t|H_1)},
\]

called the likelihood ratio for simple hypotheses \( H_0 \) and \( H_1 \). Requiring \( r > c \) gives maximum purity for a given efficiency.

N.B. any monotonic function of \( r \) is just as good.
Constructing a test statistic

Example: \( H_0 = e^+e^- \rightarrow WW \rightarrow \text{hadrons} \) (usually four jets)
\[
H_1 = e^+e^- \rightarrow qq \rightarrow \text{hadrons} \quad (\text{usually two jets})
\]

For each event measure \( \vec{x} = (x_1, \ldots, x_n) \).

According to Neyman–Pearson, to select WWs we should cut on
\[
t(\vec{x}) = \frac{f(\vec{x}|H_0)}{f(\vec{x}|H_1)},
\]
but we need to know \( f(\vec{x}|H_0) \) and \( f(\vec{x}|H_1) \).

In practice, get these from Monte Carlo event generator:

- Generate events, for each, obtain \( \vec{x} \) and enter into
  \( n \)-dimensional histogram. If e.g. \( M \) bins per component,
  total number of cells in \( \vec{x} \)-space = \( M^n \)

Approximate \( f(\vec{x}|H) \) by probability to be in corresponding cell,
i.e. determine \( M^n \) parameters. But \( n \) is potentially large!

\( \Rightarrow \) prohibitively large number of cells to populate with MC data.

Compromise solution:

- Make Ansatz for form of \( t(\vec{x}) \) with fewer parameters;
- determine the parameters (e.g. using MC) to give best
discrimination between \( H_0 \) and \( H_1 \).
Linear test statistic

Ansatz: \[ t(\vec{x}) = \sum_{i=1}^{n} a_i x_i = \vec{a}^T \vec{x} \]

A choice of \( \vec{a} \) gives certain pdfs \( g(t|H_0), g(t|H_1) \).

Choose the \( a_i \) to maximize ‘separation’ between \( g(t|H_0), g(t|H_1) \).

\[ \rightarrow \text{Must define ‘separation’.} \]

We have the expectation values and covariances,
\[
(\mu_k)_i = \int x_i f(\vec{x}|H_k) \, d\vec{x},
\]
\[
(V_k)_{ij} = \int (x - \mu_k)_i (x - \mu_k)_j f(\vec{x}|H_k) \, d\vec{x},
\]
\[ k = 0, 1 \quad \text{(hypothesis)}, \]
\[ i, j = 1, \ldots, n \quad \text{(component of} \; \vec{x}). \]

Similarly for mean and variance of \( t(\vec{x}) \),
\[
\tau_k = \int t(\vec{x}) f(\vec{x}|H_k) \, d\vec{x} = \vec{a}^T \vec{\mu}_k, 
\]
\[
\Sigma_k^2 = \int (t(\vec{x}) - \tau_k)^2 f(\vec{x}|H_k) \, d\vec{x} = \vec{a}^T V_k \vec{a}. 
\]

We should require:

\[ \text{large } |\tau_0 - \tau_1|, \]
\[ \text{small } \Sigma_0^2, \Sigma_1^2 \quad \text{(pdfs tightly concentrated about their means).} \]
Fisher defines as a measure of separation

$$J(\mathbf{\tilde{a}}) = \frac{(\tau_0 - \tau_1)^2}{\Sigma_0^2 + \Sigma_1^2}.$$ 

The numerator of $J(\mathbf{\tilde{a}})$ is

$$(\tau_0 - \tau_1)^2 = \sum_{i,j=1}^{n} a_i a_j (\mu_0 - \mu_1)_i (\mu_0 - \mu_1)_j$$

$$= \sum_{i,j=1}^{n} a_i a_j B_{ij} = \mathbf{\tilde{a}}^T \mathbf{B} \mathbf{\tilde{a}}.$$

The denominator is

$$\Sigma_0^2 + \Sigma_1^2 = \sum_{i,j=1}^{n} a_i a_j (V_0 + V_1)_{ij} = \mathbf{\tilde{a}}^T \mathbf{W} \mathbf{\tilde{a}}.$$

This gives

$$J(\mathbf{\tilde{a}}) = \frac{\mathbf{\tilde{a}}^T \mathbf{B} \mathbf{\tilde{a}}}{\mathbf{\tilde{a}}^T \mathbf{W} \mathbf{\tilde{a}}} = \frac{\text{separation between classes}}{\text{separation within classes}}.$$

Set $\frac{\partial J}{\partial a_i} = 0 \Rightarrow \mathbf{\tilde{a}} \propto \mathbf{W}^{-1}(\mathbf{\bar{\mu}}_0 - \mathbf{\bar{\mu}}_1)$

This defines Fisher's linear discriminant function, determined up to a scale factor for $\mathbf{\tilde{a}}$.

The Fisher discriminant (continued)

We can generalize $t(\vec{x})$ to be

$$t(\vec{x}) = a_0 + \sum_{i=1}^{n} a_i x_i.$$ 

Use the arbitrary scale and the offset $a_0$ to fix $\tau_0, \tau_1$.

Then maximizing $J(\vec{a}) = \frac{(\tau_0 - \tau_1)^2}{\Sigma_0^2 + \Sigma_1^2}$ means minimizing

$$\Sigma_0^2 + \Sigma_1^2 = E_0[(t - \tau_0)^2] + E_1[(t - \tau_1)^2]$$

(index shows hypothesis for expectation value)

→ Maximizing Fisher’s $J(\vec{a})$ is a type of least squares principle.
The Fisher discriminant for Gaussian $\vec{x}$

Suppose $f(\vec{x} | H_k)$ is multivariate Gaussian with mean values

\[
\begin{align*}
\vec{\mu}_0 & \text{ for } H_0, \\
\vec{\mu}_1 & \text{ for } H_1,
\end{align*}
\]

and covariance matrices $V_0 = V_1 \equiv V$ for both.

The Fisher discriminant (with an offset) is

\[
t(\vec{x}) = a_0 + (\vec{\mu}_0 - \vec{\mu}_1)^T V^{-1} \vec{x}.
\]

Recall the likelihood ratio (maximum purity for given efficiency):

\[
r = \frac{f(\vec{x} | H_0)}{f(\vec{x} | H_1)}
\]

\[
= \exp\left[-\frac{1}{2}(\vec{x} - \vec{\mu}_0)^T V^{-1} (\vec{x} - \vec{\mu}_0) + \frac{1}{2}(\vec{x} - \vec{\mu}_1)^T V^{-1} (\vec{x} - \vec{\mu}_1)\right] 
\]

\[
\propto e^t
\]

That is, $t \propto \log r + \text{const}.$ (monotonic) so for this case,

\[
\Rightarrow \text{ Fisher discriminant equivalent to likelihood ratio.}
\]

\textbf{N.B.} for $\vec{x}$ following other pdfs, this no longer holds.
The Fisher discriminant for Gaussian $\mathbf{x}$ (continued)

Multivariate Gaussian $\mathbf{x}$ with equal covariance matrices also gives a simple expression for posterior probabilities, e.g.

$$P(H_0|\mathbf{x}) = \frac{f(\mathbf{x}|H_0)\pi_0}{f(\mathbf{x}|H_0)\pi_0 + f(\mathbf{x}|H_1)\pi_1} \quad \text{Bayes’ theorem}$$

$$= \frac{1}{1 + \frac{\pi_1}{\pi_0 r}}$$

For a particular choice of the offset $a_0$ this can be written as

$$P(H_0|\mathbf{x}) = \frac{1}{1 + e^{-t}} \equiv s(t),$$

which is the logistic sigmoid function:
Neural networks (1)

Used in neurobiology, pattern recognition, financial forecasting . . . here, neural nets are just a type of test statistic.

Suppose we take \( t(\vec{x}) \) to have the form

\[
t(\vec{x}) = s \left( a_0 + \sum_{i=1}^{n} a_i x_i \right)
\]

where \( s(u) = (1 + e^{-u})^{-1} \) (the ‘activation function’)

This is the **single-layer perceptron**.

\( s(\cdot) \) is monotonic \( \Rightarrow \) equivalent to linear \( t(\vec{x}) \).

---

G. Cowan – Computing and Statistical Data Analysis – University of London Postgraduate Lectures
Generalize this to the **multilayer perceptron**:

\[
    t(x) = s \left( a_0 + \sum_{i=1}^{m} a_i h_i(x) \right),
\]

where the \( h_i \) are functions of the nodes in the previous layer,

\[
    h_i(x) = s \left( w_{i0} + \sum_{j=1}^{n} w_{ij} x_j \right).
\]

\( a_i, w_{ij} \) = weights (connection strengths)

Easy to generalize to arbitrary number of layers.

Feed-forward net: values of a node depend only on earlier layers, usually only on previous layer → ‘network architecture’

More nodes → neural net gets closer to optimal \( t(x) \), but more parameters need to be determined.
Neural networks (3)

Parameters usually determined by minimizing an error function,

\[ \mathcal{E} = E_0[(t - t^{(0)})^2] + E_1[(t - t^{(1)})^2], \]

where \( t^{(0)} \), \( t^{(1)} \) are target values, e.g. 0 and 1 for logistic sigmoid, cf. least squares principle with Fisher discriminant.

In practice, replace expectation values by averages of training data from Monte Carlo. (Adjusting parameters = network ‘learning’.)

In general this can be tricky; fortunately, programs like JETNET do it for you, e.g. with ‘error back-propogation’.

For more information see

C.M. Bishop, *Neural Networks for Pattern Recognition*,
Clarendon Press, Oxford (1995);
Neural networks (4)

An example with WW event selection

(Garrido, Juste and Martinez, ALEPH 96-144)

The input variables:

Shaded histograms: WW (signal)

Open histograms: $q\bar{q}$ (background)

The neural network output:
Choosing the input variables

Why not use all of the available input variables?

Fewer inputs $\rightarrow$ fewer parameters to be adjusted,
$\rightarrow$ parameters better determined for finite training data.

Some inputs may be highly correlated $\rightarrow$ drop all but one.

Some inputs may contain little or no discriminating power between the hypotheses $\rightarrow$ drop them.

NN exploits higher moments of joint pdf $f(\vec{x}|H)$, but these may not be well modeled in training data.

$\rightarrow$ better to have simpler $t(\vec{x})$ where you can ‘understand what it’s doing’.

Recall that the purpose of the statistical test is usually to select objects for further study; e.g. select WW events, then measure their properties (e.g. particle multiplicity).

$\Rightarrow$ avoid input variables that are correlated with the properties of the selected objects which you want to study.
(Not always easy; correlations may not be well known.)
Lecture 4 summary

- **Statistical tests**: test to what extent data stand in agreement with predicted probabilities, i.e. hypotheses.
- **Test statistics**: reduce vector $\mathbf{x}$ to a single (or few) component function $t(\mathbf{x})$.
- **The ingredients of a test**: critical region, significance level, power, (related to efficiency, purity).
- **The Neyman-Pearson lemma**: gives cut region with maximum purity for a given efficiency.
- **Constructing a test statistic**: likelihood ratio best, but usually need to determine too many parameters.
- **Alternative Ansätze for statistics**:
  - Fisher discriminant function (linear)
  - Neural network (nonlinear)