Stats 217hf: Class 2

1 An example of DS Calculus: “Object Recognition”

Topics:
- Inclusion Exclusion
- Commonalities
- The Möbius transform
- Object Recognition
- Combination
- Conflict

1.1 Consider a problem

Stored Library of \( n \) objects on which you have \( k \) binary attributes (finger print identification).

You are given 1 query object. You are asked to identify which object it is.

1.1.1 What this could be about?

This could be as simple as identifying:
- \( q : 01010 \)
- \( i : 01010 \) and so this is definitely object \( q \)

But instead consider query probabilities that cases are true.

Simplify the case of 1 attribute.

- Query measured \( p_0 = 0.2 \)
- Object 1 has \( p_1 = 0.8 \)
- Object 2 has \( p_2 = 0.2 \)

1.2 Difference between bayes and not

Consider instead alternative system:

- Query measured \( p_0 = 0.2, q_0 = 0.3, r_0 = 0.5 \)
- Clearly a multinomial rather than a binomial?
1.3 What is the “match” probability for the set \{ object = 1 \}

The probabilities are considered subjective.
You would choose to match by flipping coins with a given heads/tails probability.

- \( P(\#1 \text{ matches}) = p_0 p_1 + q_0 q_1 \)
- \( P(\#2 \text{ matches}) = p_0 p_2 + q_0 q_2 \)
- \( P(\#1 \& \#2 \text{ match}) = p_0 p_1 p_2 + q_0 q_1 q_2 \)

The flipping game is associated stochastic probability. This is a “associated” or \( A- \) probability problem.

In the Dempster-Shafer calculus there is an inclusion/exclusion argument.

1.4 Bernoulli’s Latin Problem Part 4

Consider a mixture distribution, a population which contains points of characteristics 0 and 1 with real probabilities \( p \) and \( 1 - p = q \).

We are given two machines, they report:

- Machine 1, \( p > .8 \) with 100% confidence
- Machine 2, \( q > .3 \) with 100% confidence.

I don’t know what these two machines are thinking, they are clearly lying. Consider a machine that tells us a lower bound for ratio percentage, but not an exact probability.

1.5 A subtraction of commonalities rule

Back to the Object yes/no recognition

\[
\text{Commonality} \left( \{1\} \right) = \sum_{S \supseteq \{1\}} \text{Mass} \left( S \right)
\]

Consider the alternator problem:

\[
\text{Mobius Transform} \begin{align*}
\text{Mass} \left( S \right) & = \sum_{T \supseteq S} \left( -1 \right)^{|t| - |s|} \text{Commonality} \left( T \right)
\end{align*}
\]

If you have a function over sets, a distribution over sets, we can use a relationship to calculate the probabilities of exact matches.

So however probabilities were generated over all of the combinations of sets of returns, we can calculate the probabilities of individual correctness at the end.

A theory that says, the Commonality_{Joining A,B} \left( S \right) = \text{Commonality}_A \left( S \right) \times \text{Commonality}_B \left( S \right).

1.6 A motivation

These statements were a motivation of the basic problem of “object recognition” analysis. In a chapter written in a book is the correct, complete \( p, q, r \) description. Art can send out at least 2 chapters from the book that describes all of these models. There is a particular chapter at the end of the book that squashes the mathematical approximation. Next week we’ll go over the chapter 2 material. The plan for the second half of the meeting tonight is to take up the astrophysics example.
2 The Astrophysics Example

Case \( n = 1 \), Converts \( X_1, X_2, X_3 \) come from independent poissons, \( L_1, L_2, L_3 \) where:

- \( X_1 \sim \text{Poisson} (BU + E) \)
- \( X_2 \sim \text{Poisson} (aE) \)
- \( X_3 \sim \text{Poisson} (bU) \)

A total likelihood in this model:

\[
P(X_B, U, E) = \frac{(BU + E)^{X_1} e^{-(BU + E)}}{X_1!} \times \frac{(aE)^{X_2} e^{-(aE)}}{X_2!} \times \frac{(aE)^{X_3} e^{-(BU)}}{X_3!}
\]

Splitting up the Binomial and using flat positive priors for \( B, U, E \) we have an interesting non linear equation:

\[
P(U, B, U | X) \propto \sum_{j=0}^{X_1} \left( \begin{array}{c} X_1 \\ j \end{array} \right) (BU)^j E^{X_1-j} \times E^{X_2} U^{X_3} e^{-(BU + E) - aE - bU}
\]

We have \( \int_0^\infty E^n e^{-ax} dE = \frac{\Gamma(n+1)}{a^{n+1}} \)

\[
P(U, B | X) \propto \sum_{j=0}^{X_1} \left( \begin{array}{c} X_1 \\ j \end{array} \right) B^j U^j \frac{\Gamma(X_1 - j + X_2 + 1)}{(a + 1)^{X_1-j+X_2+1}} \times e^{-BU-bU}
\]

The next integral is probably even easier:

\[
P(B | X) \propto \sum_{j=0}^{X_1} \left( \begin{array}{c} X_1 \\ j \end{array} \right) B^j \frac{\Gamma(X_1 - j + X_2 + 1) \Gamma(j + X_3 + 1)}{(a + 1)^{X_1-j+X_2+1} (b + j)^{j+X_3+1}}
\]

Well, there it goes, flat priors of course. Is this all they wanted? Unfortunately these individual probabilities can’t be calculate very well for \( X_1 \) large. For \( X_1 < 10 \) this might be rather easy.
2.1 Posterior Output

Here’s an posterior distribution, assuming $a = 1, b = 1, X_1 = 8, X_2 = 4, X_3 = 4$ then the Posterior result:

![Posterior Density, from flat priors](image1)

$X_1 = 8, X_2 = 4, X_3 = 4, a = 1, b = 1$

So we’re talking about Gamma like output. We could get the 95% confidence interval afterwards.

Now if $X_1 = 17, X_2 = 8, X_3 = 8, a = 1, b = 1$ we see: Of course, I haven’t come close to

![Posterior Density, from flat priors](image2)

$X_1 = 17, X_2 = 8, X_3 = 8, a = 1, b = 1$

calculating a confidence interval, the decision, risk, loss necessary, but clearly for any gamma priors to the $B, U, E$ we’re going to have an analytic solution to what we’re looking for.
2.2 A Reasonable Example From Paul Edlefsen

2.3 Computer Code

This completely analytic input:

```r
CalcLike <- function(B, X_1, X_2, X_3, a, b) {
  Retter <- 0
  for (j in 0:X_1) {
    Retter <- Retter + exp(LPart1(B, j, X_1, X_2, X_3, a, b))
  }
  return(Retter)
}

LPart1 <- function(B, j, X_1, X_2, X_3, a, b) {
  Returner <- lgamma(X_1 + 1) - lgamma(j + 1) - lgamma(X_1 - j + 1) +
              lgamma(X_1 - j + X_2 + 1) + lgamma(j + X_3 + 1) -
              (X_1 - j + X_2 + 1) * log(a + 1) - (j + X_3 + 1) * log(b+B) + j * log(B)
  return(Returner)
}

LPart1(.4, 1, X_1, X_2, X_3, a,b)
X_1 <- 8
X_2 <- 4
X_3 <- 4
a <- 1
b <- 1
BBSet <- 1:200 / 200 * 10
PBSet <- BBSet * 0
for (ii in 1:length(BBSet)) {
  PBSet[ii] <- CalcLike(BBSet[ii], X_1, X_2, X_3, a, b)
}

PBSet <- PBSet / sum(PBSet * (BBSet[2] - BBSet[1]))
plot(PBSet~BBSet, type="l", col="black", main="Posterior Density, from flat priors")
```