Statistical properties of algebraic combinations of distributions for property testing.

Nuño Sempere Quinn Dougherty

Abstract

This document outlines some properties about algebraic combinations of distributions. It is meant to facilitate property tests for Squiggle, an estimation language for forecasters. So far, we are focusing on the means, the standard deviation and the shape of the pdfs.

The academic keyword to search for in relation to this document is "algebra of random variables". Squiggle doesn't yet support getting the standard deviation, denoted by σ, but such support could yet be added.

Means and standard deviations

Sums

$$mean(f+g) = mean(f) + mean(g)$$

$$\sigma(f+g) = \sqrt{\sigma(f)^2 + \sigma(g)^2}$$

In the case of normal distributions,

$$mean(normal(a,b) + normal(c,d)) = mean(normal(a+c, \sqrt{b^2+d^2}))$$

Subtractions

$$mean(f-g) = mean(f) - mean(g)$$

$$\sigma(f-g) = \sqrt{\sigma(f)^2 + \sigma(g)^2}$$

Multiplications

$$mean(f \cdot g) = mean(f) \cdot mean(g)$$

$$\sigma(f \cdot g) = \sqrt{(\sigma(f)^2 + mean(f)) \cdot (\sigma(g)^2 + mean(g)) - (mean(f) \cdot mean(g))^2}$$

Divisions

Divisions are tricky, and in general we don't have good expressions to characterize properties of ratios. In particular, the ratio of two normals is a Cauchy distribution, which doesn't have to have a mean.

Probability density functions (pdfs)

Specifying the pdf of the sum/multiplication/... of distributions as a function of the pdfs of the individual arguments can still be done. But it requires integration. My sense is that this is still doable, and I (Nuño) provide some *pseudocode* to do this.

Sums

Let f, g be two independently distributed functions. Then, the pdf of their sum, evaluated at a point z, expressed as $(f+g)(z)$, is given by:

$$(f+g)(z) = \int_{-\infty}^{\infty} f(x) \cdot g(z-x)\, dx$$

See a proof sketch here

Here is some pseudocode to approximate this:

```
// pdf1 and pdf2 are pdfs,
// and cdf1 and cdf2 are their corresponding cdfs

let epsilonForBounds = 2**(-16)
let getBounds  = cdf => {
  let cdf_min = -1
  let cdf_max = 1
  let n=0
  while(
    (
      cdf(cdf_min) > epsilonForBounds ||
      ( 1 - cdf(cdf_max) ) > epsilonForBounds
    ) &&
    n < 10
  ){
    if(cdf(cdf_min) > epsilonForBounds){
      cdf_min = cdf_min * 2
    }
    if((1-cdf(cdf_max)) > epsilonForBounds){
      cdf_max = cdf_max * 2
    }
```

```
  }
  return [cdf_min, cdf_max]
}

let epsilonForIntegrals = 2**(-16)
let pdfOfSum = (pdf1, pdf2, cdf1, cdf2, z) => {
  let bounds1 = getBounds(cdf1)
  let bounds2 = getBounds(cdf2)
  let bounds = [
    Math.min(bounds1[0], bounds2[0]),
    Math.max(bounds1[1], bounds2[1])
  ]

  let result = 0
  for(let x = bounds[0]; x=x+epsilonForIntegrals; x<bounds[1]){
      let delta = pdf1(x) * pdf2(z-x)
      result = result + delta * epsilonForIntegrals
  }
  return result
}
```

Cumulative density functions

TODO

Inverse cumulative density functions

TODO

To do:

- Provide sources or derivations, useful as this document becomes more complicated
- Provide definitions for the probability density function, exponential, inverse, log, etc.
- Provide at least some tests for division
- See if playing around with characteristic functions turns out anything useful