Propagating Measurement Uncertainty Via Data Processing Algorithms: How to Take Higher Moments Into Account

Olga Kosheleva[1] and Vladik Kreinovich[2]

Departments of [1]Teacher Education and [2]Computer Science
University of Texas at El Paso, El Paso TX 79968, USA
olgak@utep.edu, vladik@utep.edu

1. Need for data processing uncertainty

- In many real-life situations, we cannot directly measure the quantity of interest; for example:
  - we cannot directly measure the distance to a faraway star or the amount of oil in an oilfield;
  - we cannot directly measure tomorrow's temperature, etc.
- Since we cannot estimate such quantity y directly, we need to estimate it indirectly:
  - we measure quantities $x_1, \ldots, x_n$ which are related to y by a known dependence $y = f(x_1, \ldots, x_n)$, and
  - we use the results $\widetilde{x}_1, \ldots, \widetilde{x}_n$ of these measurements to compute desired estimate $\widetilde{y} = f(\widetilde{x}_1, \ldots, \widetilde{x}_n)$.

2. Uncertainty propagation: a problem

- Measurement are never absolutely accurate.
- Measurement results $\widetilde{x}_i$ are usually somewhat different from the actual (unknown) values x_i of the measured quantities.
- Hence, the result $\widetilde{y}$ of data processing is, in general, somewhat different from the ideal value y.
- In many practical situations, it is important to know how accurately $\widetilde{y}$ approximates y.
- For example, suppose that we have estimated that the potential oilfield contains 150 million tons of oil; then:
  - if the estimation accuracy is ± 50, this is great, we can start exploiting;
  - but if the estimation accuracy is ± 200, maybe there is no oil at all, more measurement are needed before we invest money.

3. Possibility of linearization

- Usually, measurement errors $\Delta x_i \stackrel{\text{def}}{=} \widetilde{x}_i - x_i$ are reasonable small.
- So, we can safely ignore terms which are quadratic (or higher order) with respect to Δx_i.
- For example, even when $\Delta x_i \approx 10\%$, we have $(\Delta x_i)^2 = 1\% \ll 10\%$.
- So, we can expand the difference $\Delta y \stackrel{\text{def}}{=} \widetilde{y} - y$ in Taylor series in terms of Δx_i and only keep linear terms in this expansion:

$$\Delta y = f(\widetilde{x}_1, \ldots, \widetilde{x}_n) - f(\widetilde{x}_1 - \Delta x_1, \ldots, \widetilde{x}_n - \Delta x_n) \approx$$

$$\sum_{i=1}^{n} c_i \cdot \Delta x_i, \text{ where } c_i \stackrel{\text{def}}{=} \frac{\partial f}{\partial x_i}_{|x_i = \widetilde{x}_i}.$$

4. It is often assumed that the measurement errors are normally distributed

- For many measuring instruments, the probability distribution of their measurement error is Gaussian.
- This empirical fact has a good explanation – for many measuring instruments:
  - all major error sources are taken care of, so
  - all that remains is a large number of small independent error sources.
- According to the Central Limit Theorem, the joint effect of a large number of small independent random variables is close to Gaussian.
- A linear combination of independent Gaussian variables Δx_i is also Gaussian.
- A Gaussian distribution is uniquely determined by its first two moments.

5. It is often assumed that the measurement errors are normally distributed (cont-d)

- For measuring instruments, first moment – bias – can be eliminated by calibration.
- So, it is enough to know the second moment, which is in this case σ^2.
- So, we get the usual GUM formula for the independent case:

$$\sigma^2 = \sum_{i=1}^{n} c_i^2 \cdot \sigma_i^2.$$

6. Why and how should we take into account higher-order moments

- As we mentioned, for many measuring instruments, the measurement error is normally distributed.
- However, for almost half of the real-life measurement instruments the distribution is different from Gaussian.
- Reason: carefully calibrated measuring instruments are often very expensive to be useful in simple situations.
- So practitioners use cheaper and not so well-calibrated instruments.
- For non-Gaussian distributions, it is not enough to take into account the first two moments.
- We need to take into account higher moments as well: skewness to describe asymmetry, kurtosis to describe heavy tails, etc.
- How can we use GUM-style ideas to propagate such higher moments via data processing algorithms?

7. How to we take into account higher-order moments (cont-d)

- Natural solution – use *cumulants* κ_ℓ, defined as

$$\ln(E[\exp(\mathrm{i}\cdot t\cdot\eta)]) = \sum_{\ell=1}^{\infty} \kappa_\ell \cdot \frac{(\mathrm{i}\cdot t)^\ell}{\ell!} = \mathrm{i}\cdot\mu\cdot t - \sigma^2\cdot\frac{t^2}{2} + \ldots$$

- It is known that each κ_ℓ is an easy-to-compute rational function of the first ℓ moments $M_1, \ldots, M_\ell$.
- Vice versa, each moment M_ℓ is an easy-to-compute rational function of the first ℓ cumulants.
- Cumulants are additive: if η and η' are independent, then $\kappa_\ell(\eta+\eta') = \kappa_\ell(\eta) + \kappa_\ell(\eta')$; thus:

$$\kappa_\ell(\Delta y) = \sum_{i=1}^{n} c_i^\ell \cdot \kappa_\ell(\Delta x_i).$$

8. Need for many efficient uncertainty estimation

- Sometimes, GUM-style error propagation works well.
- However, often, data processing algorithms are proprietary – "black boxes".
- In this case, to estimate each c_i, we need to use numerical differentiation

$$c_i \approx \frac{f(\widetilde{x}_1, \ldots, \widetilde{x}_{i-1}, \widetilde{x}_i + h, \widetilde{x}_{i+1}, \ldots, \widetilde{x}_n) - \widetilde{y}}{h}.$$

- For n inputs, we need to call f n times.
- Computing f often takes long time.
- When n is in thousands, calling f n times takes very long time.

9. Monte-Carlo techniques

- We several (K) times simulate measurement errors $\Delta x_i^{(k)}$.
- We use these values to compute

$$\Delta y^{(k)} \stackrel{\text{def}}{=} \widetilde{y} - f(\widetilde{x}_1 - \Delta x_1^{(k)}, \ldots, \widetilde{x}_n - \Delta x_n^{(k)}).$$

- We estimate σ^2 as sample variance $\sigma^2 \approx \dfrac{1}{K} \cdot \sum_{k=1}^{K} (\Delta y^{(k)})^2$.
- In statistical simulations, after K iterations, we get accuracy $1/\sqrt{K}$.
- In estimating accuracy of accuracy, 20% is good: 10% or 12% measurement accuracy is about the same.
- To get 20%, we need $K = 25$.
- So, when $n > 25$, Monte-Carlo methods are faster.
- It is desirable to extend Monte-Carlo methods to higher moments as well.

10. Related challenge and its solution

- It is easy to simulate Gaussian variables,
- But how can we simulate non-Gaussian distributions with known values of higher-order moments?
- Our answer: it is known that every two continuous transformations can be obtained from each other by some bijection of real line.
- So, each distribution is equal to $F(\eta)$ for some function F, where $\eta \sim N(0,1)$.
- Each continuous function can be approximated, with any given accuracy, by a polynomial.
- Thus, we can use $P(\eta)$ for some polynomial $P(x)$.
- To fit d desired moments, we need d parameters – i.e., we need

$$P(\eta) = a_0 + a_1 \cdot \eta + \ldots + a_{d-1} \cdot \eta^{m-1}.$$

11. Related challenge and its solution (cont-d)

- Then, each moment is $E[(P(\eta))^i]$ is the expected value of some polynomial of η.
- Such value is easy to compute since:
  - for odd j, we have $E[\eta^j] = 0$, and
  - for even j, we have $E[\eta^j] = (j-1)!! \stackrel{\text{def}}{=} 1 \cdot 3 \cdot \ldots \cdot (j-1)$.
- Thus, we can easily compute all the moments for each polynomial.
- For example, to get given first three moments M_i, we take

$$P(\eta) = a_0 + a_1 \cdot \eta + a_2 \cdot \eta^2.$$

- Then, we get $M_1 = a_0 + a_2$,

$$M_2 = E[(a_0 + a_1 \cdot \eta + a_2 \cdot \eta^2)^2] = a_0^2 + a_1^2 + 3a_2^2 + 2a_0 \cdot a_2 \text{ and}$$

$$M_3 = a_0^3 + 15a_2^3 + 3a_0 \cdot a_1^2 + 3a_0^2 \cdot a_2 + 9a_0 \cdot a_2^2 + 3a_1^2 \cdot a_2.$$

- We solve this system of equations, find $a_0, \ldots$ for each Δx_i.

12. Related challenge and its solution (cont-d)

- Then, we several times simulate $\Delta x_i^{(k)}$, and we get several case of the resulting $\Delta y^{(k)}$.
- Then, we can use sample moment as an estimate of the desired moment of Δy.

13. What if we need to take into account terms non-linear in Δx_i?

- To get more accurate estimates of accuracy, we may need to also take into account terms quadratic in Δx_i.
- In this case, the expression for Δy includes higher order terms in Δx_i.
- If we only know first d moments of Δx_i, this is not enough to estimate even the second moment of Δx_i.
- Indeed, for this, we need to know the moments of order $2d$ – and we do not know them.
- Solution: similarly to the previous case, find the coefficients of each polynomial $P_i(\eta_i)$ that has the same d moments as Δx_i.
- Then, Δy is a polynomial of η_i, so $(\Delta y)^j$ is also such a polynomial.
- Thus, the j-th moment of Δy can be easily computed.

14. What if we have fuzzy uncertainty?

- Often, we have expert estimates $\widetilde{x}_i$ instead of measurement results.
- Expert can often gauge accuracy of their estimates – but not in precise terms.
- What they can do is:
  - for each real number Δx_i,
  - mark, on the scale $[0, 1]$, the degree $\mu_i(\Delta x_i)$ to which error Δx_i is possible.
- The resulting function $\mu_i(\Delta x_i)$ is known as *membership function* or, alternatively, a *fuzzy set*.
- How can we propagate such fuzzy uncertainty through data processing algorithm?

15. What if we have fuzzy uncertainty (cont-d)

- A natural idea is to transform it to probability density.
- 0 degree means not possible – so probability is 0.
- The higher degree, the larger the probability.
- The simplest such transformation is linear: $\rho_i(\Delta x_i) = c \cdot \mu_i(\Delta x_i)$.
- The overall probability should be 1, $\int \rho_i(x)\,dx = 1$, so

$$\rho_i(\Delta x_i) = \frac{\mu_i(\Delta x_i)}{\int \mu_i(x)\,dx}.$$

- Then, we can apply, e.g., Monte-Carlo method, and get several simulate values $\Delta y^{(k)}$.
- The resulting histogram is a good approximation to the pdf $\rho(\Delta y)$.
- Then, we can reconstruct the membership function $\mu(\Delta y)$ for Δy by an inverse transformation $\mu(\Delta y) = \dfrac{\rho(\Delta y)}{\max\limits_z \rho(z)}$.

16. Acknowledgments

This work was supported in part by:

- National Science Foundation grants 1623190, HRD-1834620, HRD-2034030, and EAR-2225395;
- AT&T Fellowship in Information Technology;
- a grant from the Hungarian National Research, Development and Innovation Office (NRDI).

The authors are greatly thankful to Igor Zakharov for valuable discussions.