Lecture 3: Which Fuzzy Techniques?

1 What We Study in This Lecture

We need to select fuzzy techniques. In the previous lectures, we explained why fuzzy techniques are a reasonable tool for designing explainable AI, and we had a brief overview of fuzzy techniques. However, as we have learned, there are many different versions of fuzzy techniques. So, we need to decide which versions of these techniques are most appropriate for explainable AI.

What exactly do we need to select? In each stage of the fuzzy techniques, we need to make a selection.

We need to select an interpolation procedure. First, we need to find the membership functions corresponding to the imprecise natural-language words like "negligible" that experts use in their rules. For this purpose, we need to select an interpolation procedure. In principle:

- we can have many different interpolation procedures,
- since there are many curves that you can draw that go through two given points (x_1, y_1) and (x_2, y_2) on the plane.

The simplest of these curves is the straight line, which corresponds to linear functions and linear interpolation – and this is what we used in our example. However, simplest is not always the best – and we will see examples of that in this lecture.

We need to select "and"- and "or"operations. After that, we need to select "and"- and "or"-operations. For each of these two classes of operations, we had two examples – but there are many others. It is known that in different applications, different "and"- and "or"-operations work best.

For example, several decades ago, when Stanford researchers worked on the designing an expert system for diagnosing blood diseases, they spent a lot of effort finding "and"- and "or"-operations that best reflects reasoning of medical doctors.

- At first, the researchers thought that they have discovered the general laws of human reasoning.
- However, when they applied the same "and"- and "or"-operations when designing an expert system for geophysics, their system failed to capture what geophysicists wanted – because geophysicists think differently.

This difference is easy to explain:

- A medical doctor tries his/her best not to harm the patient. Because of this, the doctor does not make a serious decision until he/she is absolutely sure: a wrong decision can be a disaster. If in doubt, and if the situation is not an emergency, the doctor will recommend additional tests.
- In contrast, a geophysicist working for an oil company is likely to recommend exploring the area even when there may be doubts – since an unnecessary delay may help the competition get ahead. Even if sometimes, the resulting wells are dry – it is still cheaper, on average, to suffer these problems in a few cases rather than spend a lot of money on intensively studying each area.

In this lecture, we will analyze which "and"- and "or"-operations are most appropriate for applications to explainable AI.

We need to select a defuzzification procedure. Finally, we need to select the best defuzzification procedure. We have already provided arguments that the best such selection is centroid defuzzification – modified a little but if needed, so this topic has already been covered.

Which selections are better? Let us describe which selections are most appropriate for explainable AI applications.

2 Interpolation Should Be Robust

Why we need interpolation in fuzzy techniques: reminder. One of the main ideas in fuzzy techniques is that we describe each imprecise ("fuzzy") natural-language property like "small" by assigning:

- to each possible value x of the corresponding quantity,
- the degree $\mu(x)$ to which, according to the experts, the value x satisfies this property – e.g., the degree to which x is small.

To find this function, we ask the experts.

- However, we can only ask finitely many questions to the experts.
- So, by asking finitely many questions, we will be able to only find out the values $\mu\left(x^{(k)}\right)$ at finitely many points $x^{(1)}$, $x^{(2)}$, etc.

To determine the value $\mu(x)$ for all other x, we need to use interpolation.

Need for robustness. In practical applications, the value of the quantity x comes from measurements, and measurements are never absolutely accurate. Anyone who ever measured anything – be it voltage, current, blood pressure, whatever – knows that if we repeat the measurement again, we will get, in general, a slightly different value.

We want to make sure that this difference does not affect the results. For this purpose, we want to make sure that:

- if two measurement results are close, i.e., if $x \approx x'$,
- then the corresponding values of the membership function should also be close: $\mu(x) \approx \mu(x')$.

This property of not-changing-much when inputs change is known as *robustness*.

How can we describe robustness in precise terms. The above description of robustness is not precise: it uses the imprecise natural-language word "close". How can make this description more precise? For this, let us use the same ideas as we used to explain centroid defuzzification.

Suppose that:

- by asking experts, we know the values $\mu(a)$ and $\mu(b)$ for some values $a < b$, and
- we want to determine the values $\mu(x)$ for all intermediate values $x \in [a, b]$.

Let Δx denote the accuracy of our measurements. This means that:

- if we have two measurement results a and a' for which $a \le a' < a + \Delta x$,
- then these two measurement results may correspond to the same actual value of the measured quantity.

For example, suppose that the measurement accuracy is 0.1.

- If, in two consequent measurements:
  - we get the values 1.0 and 1.3,
  - this means that during the time between the two measurements, the actual value has changed.

  Indeed:
  - the difference $1.3 - 1.0 = 0.3$ between the measurement results is larger than the measurement accuracy;
  - thus, this difference cannot be explained by the measurement uncertainty only.

  In this sense, the values 1.0 and 1.3 are *distinguishable* – they allow us to distinguish between different actual values of the measured quantity.

- On the other hand, if in two consequent measurements:
  - we get the values 1.0 and 1.05,
  - then it may be that the actual value did not change.

  Indeed, in this case:

- the difference between the measurement results is smaller than the measurement accuracy $\Delta x = 0.1$;
- thus, this difference can be explained by the measurement accuracy only.

So, the values 1.0 and 1.05 are *indistinguishable* – these two measurement results may correspond to the same value of the measured quantity.

So, if we start with the value $\widetilde{x} = 1.0$ and start considering larger and larger values, then:

- at first, we get the values 1.01, 1.02, ..., 1.09 which are indistinguishable from 1.0;
- the first value which is distinguishable from 1.0 is the value

$$\widetilde{x} + \Delta x = 1 + 0.1 = 1.1.$$

In general, the values a' which are too close to a are indistinguishable from a.

- The first value which is distinguishable from a is $a + \Delta x$.
- Similarly, the first value distinguishable from $a + \Delta x$ is the value

$$(a + \Delta x) + \Delta x = a + 2 \cdot \Delta x,$$

etc.

So, in effect, we have only finitely many distinguishable values. Let us denote $x_1 = a$, $x_2 = x_1 + \Delta x$, $x_3 = x_2 + \Delta x$, ... Let us denote the last value by $x_n \approx b$.

In this case, we have:

- $x_2 = x_1 + \Delta x$,
- $x_3 = x_2 + \Delta x = (x_1 + \Delta x) + \Delta x = x_1 + 2 \cdot \Delta x$,
- $x_4 = x_3 + \Delta x = (x_1 + 2 \cdot \Delta x) + \Delta x = x_1 + 3 \cdot \Delta x$, and
- in general, $x_i = x_1 + (i - 1) \cdot \Delta x$.

Now, the interpolation problem takes the following form:

- We know the value $\mu(x_1) = \mu(a)$, and we know the value $\mu(x_n) = \mu(b)$.
- We want to find all the intermediate values $\mu(x_2)$, $\mu(x_3)$, ..., $\mu(x_{n-1})$.

In these terms, robustness means that the values $\mu(x_i)$ and $\mu(x_{i+1})$ corresponding to two nearby values x should be close to each other. In other words, we must have:

$$(\mu(a) =)\ \mu(x_1) \approx \mu(x_2), \quad \mu(x_2) \approx \mu(x_3), \quad \ldots, \mu(x_{n-1}) \approx \mu(x_n)\ (= \mu(b)).$$

In other words, the tuple

$$\ell = (\mu(x_1), \mu(x_2), \ldots, \mu(x_{n-1}))$$

formed by the left-hand sides of all the above closeness relations must be closed to the tuple

$$r = (\mu(x_2), \mu(x_3), \ldots, \mu(x_n))$$

formed by the right-hand sides of these relations.

The smaller the distance between these two tuples, the more robust the resulting membership function. So:

- if we want the most robust function,
- then we should select the intermediate values $\mu(x_2)$, $\mu(x_3)$, $\ldots$, $\mu(x_{n-1})$ for which the distance $D(\ell, r)$ between these two tuples is the smallest possible.

3 Which Interpolation Is the Most Robust

How we can find the most robust interpolation: idea. We already know the formula for the distance between the two tuples. According to this formula, the square of this distance has the form

$$D^2(\ell, r) = (\mu(x_2) - \mu(x_1))^2 + (\mu(x_3) - \mu(x_2))^2 + \ldots + (\mu(x_n) - \mu(x_{n-1}))^2.$$

For each i, to find the value $\mu(x_i)$, we need:

- to differentiate the minimized expression $D^2(\ell, r)$ with respect to the unknown $\mu(x_i)$, and
- to equate the derivative to 0.

Differentiation with respect to $\mu(x_2)$. Let us start with the case $i = 2$, when we need to find the value $\mu(x_2)$.

In the above expression, only the first two terms

$$(\mu(x_2) - \mu(x_1))^2 \text{ and } (\mu(x_3) - \mu(x_2))^2$$

depend on $\mu(x_2)$. All other terms do not depend on $\mu(x_2)$ and thus, their derivatives with respect to the unknown $\mu(x_2)$ is equal to 0. Thus, the derivative of the sum $D^2(\ell, r)$ with respect to $\mu(x_2)$ is equal to the sum of the derivatives of the first two terms in this sum.

By using chain rule to differentiate each of these two terms, we find the expression for this derivative, which should be equal to 0;

$$2(\mu(x_2) - \mu(x_1)) + 2 \cdot (\mu(x_3) - \mu(x_2)) \cdot (-1) = 0.$$

To simplify this formula, we can divide both sides by 2, thus:

$$(\mu(x_2) - \mu(x_1)) + (\mu(x_3) - \mu(x_2)) \cdot (-1) = 0.$$

Adding $\mu(x_3) - \mu(x_2)$ to both sides, we conclude that

$$\mu(x_2) - \mu(x_1) = \mu(x_3) - \mu(x_2). \tag{1}$$

Differentiation with respect to $\mu(x_3)$. Similarly, for $i = 3$, the only two terms in the sum $D^2(\ell, r)$ that depend on the value $\mu(x_3)$ are the terms

$$(\mu(x_3) - \mu(x_2))^2 \text{ and } (\mu(x_4) - \mu(x_3))^2.$$

So, the derivative of the sum $D^2(\ell, r)$ with respect to $\mu(x_3)$ is equal to the sum of the derivatives of these two terms.

Here, similarly, by using the chain rule and equating the derivative to 0, we get the following equation:

$$2 \cdot (\mu(x_3) - \mu(x_2)) + 2 \cdot (\mu(x_4) - \mu(x_3)) \cdot (-1) = 0.$$

To simplify this formula, we can divide both sides by 2, thus:

$$(\mu(x_3) - \mu(x_2)) + (\mu(x_4) - \mu(x_3)) \cdot (-1) = 0.$$

Adding $\mu(x_4) - \mu(x_3)$ to both sides of this equality, we conclude that

$$\mu(x_3) - \mu(x_2) = \mu(x_4) - \mu(x_3). \tag{2}$$

Differentiation with respect to other unknowns. Similarly, by considering the unknown value $\mu(x_4)$, we conclude that

$$\mu(x_4) - \mu(x_3) = \mu(x_5) - \mu(x_4), \tag{3}$$

and, in general, by considering the unknown value $\mu(x_i)$, we conclude that

$$\mu(x_i) - \mu(x_{i-1}) = \mu(x_{i+1}) - \mu(x_i). \tag{4}$$

What we get. From the formulas (1)–(4), we conclude that

$$\mu(x_2) - \mu(x_1) = \mu(x_3) - \mu(x_2) = \mu(x_4) - \mu(x_3) = \mu(x_5) - \mu(x_4) = \ldots =$$
$$\mu(x_i) - \mu(x_{i-1}) = \ldots$$

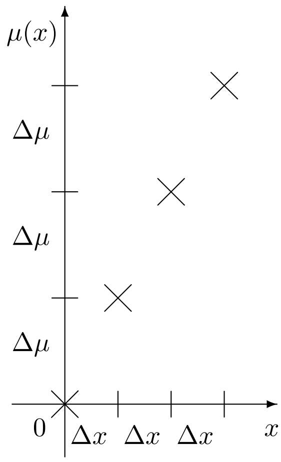
In other words, for each i, the difference $\mu(x_i) - \mu(x_{i-1})$ between the values $\mu(x_i)$ and $\mu(x_{i-1})$ of the membership function at nearby points is the same. Let us denote this common difference by $\Delta\mu$. In terms of this notation, we have:

$$\mu(x_2) - \mu(x_1) = \Delta\mu; \quad \mu(x_3) - \mu(x_2) = \Delta\mu; \ldots \mu(x_i) - \mu(x_{i-1}) = \Delta\mu; \ldots$$

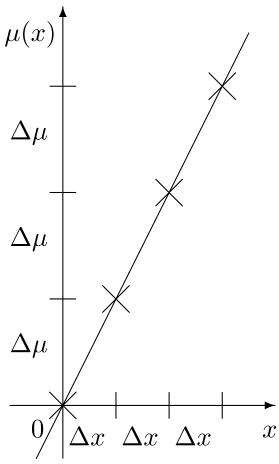
Then:

- by adding $\mu(x_1)$ to both sides of the equality $\mu(x_2) - \mu(x_1) = \Delta\mu$, we conclude that
$$\mu(x_2) = \mu(x_1) + \Delta\mu;$$
- by adding $\mu(x_2)$ to both sides of the equality $\mu(x_3) - \mu(x_2) = \Delta\mu$, we conclude that $\mu(x_3) = \mu(x_2) + \Delta\mu$; substituting the above expression for $\mu(x_2)$ into this formula, we conclude that
$$\mu(x_3) = (\mu(x_1) + \Delta\mu) + \Delta\mu = \mu(x_1) + 2 \cdot \Delta\mu;$$
- by adding $\mu(x_3)$ to both sides of the equality $\mu(x_4) - \mu(x_3) = \Delta\mu$, we conclude that $\mu(x_4) = \mu(x_3) + \Delta\mu$; substituting the above expression for $\mu(x_3)$ into this formula, we conclude that
$$\mu(x_4) = (\mu(x_1) + 2 \cdot \Delta\mu) + \Delta\mu = \mu(x_1) + 3 \cdot \Delta\mu;$$
- in general, we get
$$\mu(x_i) = \mu(x_1) + (i - 1) \cdot \Delta\mu.$$

Here, the same increase Δx in x leads to the same increase $\Delta\mu$ in μ:



Let us prove that the resulting dependence is linear. This is similar to a situation when a car goes with a constant speed. For the car, this means the distance increases linearly with time. Let us show that the dependence of $\mu(x)$ on x is also linear.



We want to find the dependence of $\mu(x)$ on x. What we have so far is a description of how both x_i and $\mu(x_i)$ depend on i: $x_i = x_1 + (i-1)\cdot\Delta x$ and $\mu(x_i) = \mu(x_1) + (i-1)\cdot\Delta\mu$. So, to find the dependence of $\mu(x)$ on x, a natural idea is:

- to use the known relation $x_i = x_1 + (i-1)\cdot\Delta x$ between x_i and i to describe i in terms of x_i, and then
- to substitute the expression for i in terms of x_i into the formula that describes $\mu(x_i)$ in terms of i.

Let us follow this idea. To find the expression of i in terms of x_i, let us:

- subtract x_1 from both sides of the equality $x_i = x_1 + (i-1)\cdot\Delta x$, and then
- divide both sides by Δx.

Thus, we conclude that
$$i - 1 = \frac{x_i - x_1}{\Delta x}.$$
Substituting the right-hand side of this equality instead of $i-1$ into the formula
$$\mu(x_i) = \mu(x_1) + (i-1)\cdot\Delta\mu,$$
we conclude that
$$\mu(x_i) = \mu(x_1) + \frac{x_i - x_1}{\Delta x}\cdot\Delta\mu,$$
i.e.,
$$\mu(x_i) = \frac{\Delta\mu}{\Delta x}\cdot x_i + \left(\mu(x_1) - \frac{x_1}{\Delta x}\cdot\Delta\mu\right),$$
i.e., the desired linear form
$$\mu(x_i) = a\cdot x_i + b,$$

where we denoted:

$$a = \frac{\Delta\mu}{\Delta x} \text{ and } b = \left(\mu(x_1) - \frac{x_1}{\Delta x} \cdot \Delta\mu\right).$$

Conclusion of this section. Our analysis shows that *the most robust interpolation is linear interpolation* – exactly what we used to determine the membership functions in our example.

Discussion. For interpolation, the most robust procedure turned out to be the same as the simplest procedure – namely, linear interpolation. However, as we will see in the next section on the example of "and"- and "or"-operations, the most robust is not always the same as the simplest.

Terminological comment.

- In the usual optimization problems (e.g., when we selected the optimal control value in the previous lecture) the unknown – that we want to determine – is a *number*.
- In this section, the unknown – that we need to determine – is a *function*, namely, the function $\mu(x)$. In other words, we use optimization to determine how the value $\mu(x)$ varies when x changes.

Such problems, in which we use optimization to determine a function, are known as problems of *variational optimization*, and the derivative of the minimized function with respect to the unknown value $f(x)$ is known as the *variational derivative.*

Related comments about notations. Derivative with respect to $f(x)$:

- is sometimes denoted differently than the usual notation $\dfrac{dy}{dx}$ for the derivative,
- namely, as $\dfrac{\delta y}{\delta f(x)}$.

The main purpose of this difference is to emphasize that the unknown is a function.

4 "And"- and "Or"-Operations Must Be Robust Too

Expert's degrees are only approximate. The expert estimate depends on a scale. If we ask the expert to estimate the degree on a scale from 0 to 5, then possible values of the resulting degree are:

$$0/5 = 0.0;\quad 1/5 = 0.2;\quad 2/5 = 0.4;\quad 3/5 = 0.6;\quad 4/5 = 0.8;\ \text{and } 5/5 = 1.0.$$

However, if we ask the same expert to estimate his/her degree on a scale from 0 to 4, then we will get different possible values:

$$0/4 = 0.0; \quad 1/4 = 0.25; \quad 2/4 = 0.5; \quad 3/4 = 0.75; \text{ and } 4/4 = 1.0.$$

Suppose that in the first scale, the expert marked 4 on a scale from 0 to 5, leading to an estimate of 0.8. However, no mark on a 0 to 4 scale will lead to the same value 0.8; the closest is the value 0.75 which corresponds to 3 on the 0 to 4 scale. The value 0.75 is close to 0.8, but different.

Similar problem occurs if we use polling: for different numbers of experts, we get different values describing the same degrees of belief.

So, we need robustness. As we have argued, the same confidence level of an expert leads, in general, to different degrees $a \neq a'$ – depending on the scale or on the number of experts.

It is therefore reasonable to require that the corresponding small difference $a' - a$ should affect the results as little as possible. In particular, for an "and"-operation, we should require that, for each b:

- if a is close to a' – which we denoted by $a \approx a'$,
- then the value $f_\&(a, b)$ should be close to $f_\&(a', b)$: $f_\&(a, b) \approx f_\&(a', b)$.

Similarly, for an "or"-operation, we should require that, for each b:

- if a is close to a' – which we denoted by $a \approx a'$,
- then the value $f_\vee(a, b)$ should be close to $f_\vee(a', b)$: $f_\vee(a, b) \approx f_\vee(a', b)$.

5 Which Is the Most Robust "And"-Operation

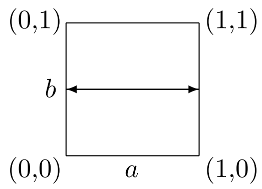
What is an "and"-operation: reminder. In our analysis, we will use the following two properties of an "and"-operation $f_\&(a, b)$:

- that it is commutative $f_\&(a, b) = f_\&(b, a)$, and
- that it coincides with the usual "and" when both a and b are equal to 0 or 1 (absolutely false or absolutely true):

$$f_\&(0, 0) = f_\&(0, 1) = f_\&(1, 0) = 0 \text{ and } f_\&(1, 1) = 1.$$

Comment. The definition of an "and"-operation also requires that this operation be associative and monotonic, but we will not need these two properties in our derivation. It should be mentioned that the resulting most robust operation will be associative and monotonic.

Let us find our what is the most robust "and"-operation. In the previous section, we have shown that the most robust dependence is linear. So, for each b, the dependence of $f_\&(a, b)$ on a must be linear. To make it clearer which of the two inputs changes, we will denote the value $f_\&(a, b)$ by $f_b(a)$.



Let us use this conclusion to derive the expression for the most robust "and"-operation.

For some values b, we can directly use this conclusion. For each value b, to perform interpolation, we need to know two different values $f_b(a_1) = f_\&(a_1, b)$ and $f_b(a_2) = f_\&(a_2, b)$ of the desired function $f_b(a) = f_\&(a, b)$. Based on our definition of an "and"-operation, we have this information only for the values $b = 0$ and $b = 1$:

- for $b = 0$, we know the values

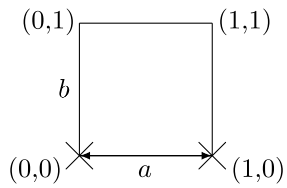
$$f_0(0) = f_\&(0, 0) = 0 \text{ and } f_0(1) = f_\&(1, 0) = 0;$$

- for $b = 1$, we know the values

$$f_1(0) = f_\&(1, 0) = 0 \text{ and } f_1(1) = f_\&(1, 1) = 1.$$

Let us therefore apply linear interpolation for these two values of b.

Case of $b = 0$. In this case, we have $f_\&(0, 0) = 0$ and $f_\&(1, 0) = 0$. So, for the dependence $f_0(a) = f_\&(a, 0)$, we have $f_0(0) = f_0(1) = 0$. We know that the dependence $y = f_0(a)$ is linear, so we can use linear interpolation formula to find the values $f_\&(a, 0)$ for all a.

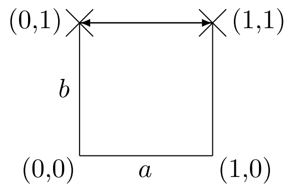


Here, $a_1 = 0$, $a_2 = 1$, $y_1 = y_2 = 0$, so the general linear interpolation formula leads to

$$f_\&(a, 0) = f_0(a) = y_1 + \frac{a - a_1}{a_2 - a_1} \cdot (y_2 - y_1) = 0 + \frac{a - 0}{1 - 0} \cdot (0 - 0).$$

Thus, we conclude that for all a, we have $f_\&(a, 0) = 0$.

Case of $b = 1$. In this case, we have $f_\&(0, 1) = 0$ and $f_\&(1, 1) = 1$. So, for the dependence $f_1(a) = f_\&(a, 1)$, we have $f_1(0) = 0$ and $f_1(1) = 1$. We know that the dependence $y = f(a)$ is linear, so we can use linear interpolation formula to find the values $f_\&(a, 1)$ for all a.



Here, $a_1 = 0$, $a_2 = 1$, $y_1 = 0$, $y_2 = 1$, so the general linear interpolation formula leads to

$$f_\&(a,1) = f_1(a) = y_1 + \frac{a - a_1}{a_2 - a_1} \cdot (y_2 - y_1) = 0 + \frac{a-0}{1-0} \cdot (1-0).$$

Thus, we conclude that for all a, we have $f_\&(a,1) = a$.

What can we do for all other values b?

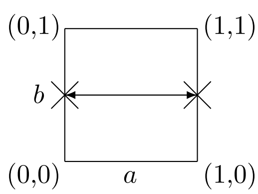
- We want to find the values $f_\&(a,b)$ corresponding to all possible a and b.
- So far, we have found the values $f_\&(a,0) = 0$ and $f_\&(a,1) = a$ corresponding to $b = 0$ and $b = 1$.

How can we find the values $f_b(a) = f_\&(a,b)$ for the values b between 0 and 1?

Since we assume that the "and"-operation is maximally robust, we can use linear interpolation. To be able to use linear interpolation, we need to know the values $f_b(a_1) = f_\&(a_1,b)$ and $f_b(a_2) = f_\&(a_2,b)$ for two different values $a_1 \neq a_2$. We can get these values by using commutativity and the already-proven properties $f_\&(a,0) = 0$ and $f_\&(a,1) = a$:

- from the fact that $f_\&(a,0) = 0$ for all a, we conclude that
$$f_\&(0,b) = f_\&(b,0) = 0;$$
- from the fact that $f_\&(a,1) = a$ for all a, we conclude that
$$f_\&(1,b) = f_\&(b,1) = b.$$

So, we have $f_b(0) = f_\&(0,b) = 0$ and $f_b(1) = f_\&(1,b) = b$. So, for the dependence $f_b(a) = f_\&(a,b)$, we have $f_b(0) = 0$ and $f_b(1) = b$. We know that the dependence $y = f_b(a)$ is linear, so we can use linear interpolation formula to find the values $f_b(a) = f_\&(a,b)$ for all a.



Here, $a_1 = 0$, $a_2 = 1$, $y_1 = 0$, $y_2 = b$, so the general linear interpolation formula leads to

$$f_\&(a, b) = f_b(a) = y_1 + \frac{a - a_1}{a_2 - a_1} \cdot (y_2 - y_1) = 0 + \frac{a - 0}{1 - 0} \cdot (b - 0).$$

Thus, we conclude that for all a, we have $f_\&(a, b) = a \cdot b$.

Conclusion of this section. The most robust "and"-operation is algebraic product.

Discussion. For interpolation, the most robust procedure turned out to be the simplest. Is this the case here as well? Not really.

Indeed, minimum is easier to perform than the product: we do not need to compute anything, we just need to decide which of the two numbers is smaller. So, this is the case when the most robust selection is *different* from the simplest one.

6 Which Is the Most Robust "Or"-Operation

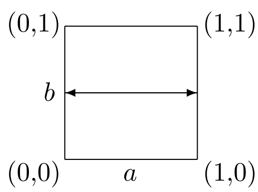
What is an "or"-operation: reminder. In our analysis, we will use the following two properties of an "or"-operation $f_\vee(a, b)$:

- that it is commutative $f_\vee(a, b) = f_\vee(b, a)$, and
- that it coincides with the usual "or" when both a and b are equal to 0 or 1 (absolutely false or absolutely true):

$$f_\vee(0, 0) = 0 \text{ and } f_\vee(0, 1) = f_\vee(1, 0) = f_\vee(1, 1) = 1.$$

Comment. The definition of an "or"-operation also requires that this operation be associative and monotonic, but we will not need these two properties in our derivation. It should be mentioned that the resulting most robust operation will be associative and monotonic.

Let us find our what is the most robust "or"-operation. In the previous section, we have shown that the most robust dependence is linear. So, for each b, the dependence of $f_\vee(a, b)$ on a must be linear. To make it clearer which of the two inputs changes, we will denote the value $f_\vee(a, b)$ by $g_b(a)$.



Let us use this conclusion to derive the expression for the most robust "or"-operation.

For some values b, we can directly use this conclusion. For each value b, to perform interpolation, we need to know two different values $g_b(a_1) = f_\vee(a_1, b)$ and $g_b(a_2) = f_\vee(a_2, b)$ of the desired function $g_b(a) = f_\vee(a, b)$. Based on our definition of an "or"-operation, we have this information only for the values $b = 0$ and $b = 1$:

- for $b = 0$, we know the values

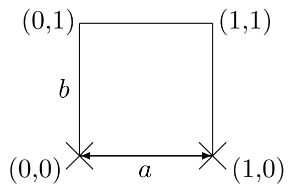
$$g_0(0) = f_\vee(0,0) = 0 \text{ and } g_0(1) = f_\vee(1,0) = 1;$$

- for $b = 1$, we know the values

$$g_1(0) = f_\vee(1,0) = 1 \text{ and } g_1(1) = f_\vee(1,1) = 1.$$

Let us therefore apply linear interpolation for these two values of b.

Case of $b = 0$. In this case, we have $f_\vee(0,0) = 0$ and $f_\vee(1,0) = 1$. So, for the dependence $g_0(a) = f_\vee(a, 0)$, we have $g_0(0) = 0$ and $g_0(1) = 1$. We know that the dependence $y = g_0(a)$ is linear, so we can use linear interpolation formula to find the values $f_\vee(a, 0)$ for all a.

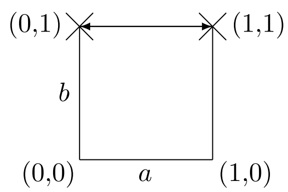


Here, $a_1 = 0$, $a_2 = 1$, $y_1 = 0$, and $y_2 = 1$, so the general linear interpolation formula leads to

$$f_\vee(a,0) = g_0(a) = y_1 + \frac{a - a_1}{a_2 - a_1} \cdot (y_2 - y_1) = 0 + \frac{a - 0}{1 - 0} \cdot (1 - 0).$$

Thus, we conclude that for all a, we have $f_\vee(a, 0) = a$.

Case of $b = 1$. In this case, we have $f_\vee(0,1) = 1$ and $f_\vee(1,1) = 1$. So, for the dependence $g_1(a) = f_\vee(a, 1)$, we have $g_1(0) = g_1(1) = 1$. We know that the dependence $y = f(a)$ is linear, so we can use linear interpolation formula to find the values $f_\vee(a, 1)$ for all a.



Here, $a_1 = 0$, $a_2 = 1$, $y_1 = y_2 = 1$, so the general linear interpolation formula leads to

$$f_\vee(a,1) = g_1(a) = y_1 + \frac{a - a_1}{a_2 - a_1} \cdot (y_2 - y_1) = 1 + \frac{a - 0}{1 - 0} \cdot (0 - 0).$$

Thus, we conclude that for all a, we have $f_\vee(a,1) = 1$.

What can we do for all other values b?

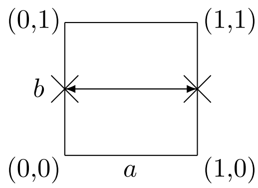
- We want to find the values $f_\vee(a,b)$ corresponding to all possible a and b.
- So far, we have found the values $f_\vee(a,0) = a$ and $f_\vee(a,1) = 1$ corresponding to $b = 0$ and $b = 1$.

How can we find the values $f_b(a) = f_\vee(a,b)$ for the values b between 0 and 1?

Since we assume that the "or"-operation is maximally robust, we can use linear interpolation. To be able to use linear interpolation, we need to know the values $g_b(a_1) = f_\vee(a_1,b)$ and $g_b(a_2) = f_\vee(a_2,b)$ for two different values $a_1 \neq a_2$. We can get these values by using commutativity and the already-proven properties $f_\&(a,0) = 0$ and $f_\&(a,1) = a$:

- from the fact that $f_\vee(a,0) = a$ for all a, we conclude that
$$f_\vee(0,b) = f_\vee(b,0) = b;$$
- from the fact that $f_\vee(a,1) = 1$ for all a, we conclude that
$$f_\vee(1,b) = f_\vee(b,1) = b.$$

So, we have $g_b(0) = f_\vee(0,b) = b$ and $g_b(1) = f_\vee(1,b) = 1$. So, for the dependence $g_b(a) = f_\vee(a,b)$, we have $g_b(0) = 0$ and $g_b(1) = b$. We know that the dependence $y = g_b(a)$ is linear, so we can use linear interpolation formula to find the values $g_b(a) = f_\vee(a,b)$ for all a.



Here, $a_1 = 0$, $a_2 = 1$, $y_1 = b$, $y_2 = 1$, so the general linear interpolation formula leads to

$$f_\vee(a,b) = g_b(a) = y_1 + \frac{a - a_1}{a_2 - a_1} \cdot (y_2 - y_1) = b + \frac{a - 0}{1 - 0} \cdot (1 - b).$$

Thus, we conclude that for all a, we have

$$f_\vee(a,b) = b + a \cdot (1 - b) = a + b - a \cdot b.$$

Conclusion of this section. The most robust "or"-operation is

$$f_\vee(a,b) = a + b - a \cdot b.$$

Discussion. In this case too – just like for "and"-operations – the most robust procedure is not the simplest. Indeed, the maximum is easier to compute than the product: we do not need to compute anything, we just need to decide which of the two numbers is larger. So, this is the case when the most robust selection is *different* from the simplest one.

7 Group Robustness vs. Individual Robustness

Robustness – as we have defined it – means robustness "on average". In the previous text, as a measure of robustness, we took the distance between the two tuples, i.e., the sum

$$D^2(\ell,r) = (\mu(x_2) - \mu(x_1))^2 + (\mu(x_3) - \mu(x_2))^2 + \ldots + (\mu(x_n) - \mu(x_{n-1}))^2$$

of the squares $(\mu(x_{i+1}) - \mu(x_i))^2$ of the differences $\mu(x_{i+1}) - \mu(x_i)$ between the values $\mu(x_{i+1})$ and $\mu(x_i)$ of the membership function at the neighboring points x_i and x_{i+1}. The smaller this sum, the more robust the interpolation. For this description of robustness, he most robust interpolation is the one for which this sum is the smallest possible.

Minimizing the sum is equivalent to minimizing the average of the squares of the differences:

$$\frac{D^2(\ell,r)}{n-1} = \frac{(\mu(x_2) - \mu(x_1))^2 + (\mu(x_3) - \mu(x_2))^2 + \ldots + (\mu(x_n) - \mu(x_{n-1}))^2}{n-1}.$$

From this viewpoint, the previous understanding of robustness means robustness "on average".

Similarly, when we selected the most robust "and"- and "or"-operations, we understood robustness "on average".

So far, we dealt with group robustness. In our analysis, we assumed that the values $f_\&(a,b)$ and $f_\&(a',b)$ are, *on average*, close to each other. This makes sense if we control, e.g., a flock of UAVs for studying weather. Even if one of them fails, we still have a good picture of the weather if most of all are successful and follow the desired trajectory.

Sometimes, we need individual robustness. In some situations, we are interested in the success of an individual object – e.g., we have a single UAV. In this case, the fact that most other UAVs – that, e.g., collect weather information in other cities – will be successful is no help if the UAV collecting weather information in our city of El Paso fails.

To deal with such situations, we do not just want to require that sum of the squares of the differences is small, we want to require that *each* difference is small, i.e., in terms of a function $f(x)$:

- if the values x and x' are close, then
- each corresponding pair $f(x)$ and $f(x')$ should also be close.

How can we describe such individual robustness. We want to make sure that if the difference $|x - x'|$ is small, then the difference $|f(x) - f(x')|$ should also be small.

Of course, the larger the difference $|x - x'|$, the larger can be the difference $|f(x) - f(x')|$ between the corresponding values of the function $f(x)$. So, the bound on $|f(x) - f(x')|$ should be proportional to $|x - x'|$, i.e., we should require that

$$|f(x) - f(x')| \leq K \cdot |x - x'|$$

for some coefficient K.

The smaller the value K, the smaller the bound on the difference

$$|f(x) - f(x')|$$

and thus, the more robust the function $f(x)$. Thus, it is reasonable to find the functions $f(x)$ for which the value K is the smallest possible.

Historical comment. The condition that the inequality $|f(x) - f(x')| \leq K \cdot |x - x'|$ must be satisfied for all x ad x' is known as the *Lipschitz condition*, after a mathematician who first used it. If a function $f(x)$ satisfies this condition, we say that this function is a K-*Lipschitz* function.

8 Which Interpolation Is the Most Individually Robust

Formulation of the problem. Let us consider the same formulation as before:

- we know the values $f(a)$ and $f(b)$, and
- we want to find the values $f(x)$ for all $x \in (a, b)$ so as to maintain maximum individual robustness.

A natural lower bound on K. The desired inequality $|f(x) - f(x')| \leq K \cdot |x - x'|$ must be held for all possible values x and x', in particular, for the values $x = a$ and $x' = b$. In this case, this inequality takes the form $|f(b) - f(a)| \leq K \cdot |b - a|$. If we divide both sides by $|b - a|$, we conclude that

$$K \geq \frac{|f(b) - f(a)|}{|b - a|}.$$

So, the value K cannot be smaller than the ratio in the right-hand side of this inequality. We will denote this ratio by

$$r = \frac{|f(b) - f(a)|}{|b - a|};$$

in terms of this ratio, we must have $K \geq r$.

What happens for linear interpolation. So far, we have studied only one interpolation algorithm: namely, linear interpolation. According to this algorithm, for each value x, we take

$$f_L(x) = f(a) + \frac{x - a}{b - a} \cdot (f(b) - f(a)).$$

This expression can be rewritten in the following equivalent form

$$f_L(x) = f(a) + \frac{f(b) - f(a)}{b - a} \cdot (x - a),$$

i.e., in the form

$$f_L(x) = f(a) + K_0 \cdot (x - a),$$

where we denoted

$$K_0 = \frac{f(b) - f(a)}{b - a}.$$

For every two values x and x', we therefore have the following expression for the difference $f_L(x) - f_L(x')$:

$$f_L(x) - f_L(x') = (f(a) + K_0 \cdot (x - a)) - (f(a) + K_0 \cdot (x' - a)) =$$

$$f(a) + K_0 \cdot x - K_0 \cdot a - f(a) - K_0 \cdot x' + K_0 \cdot a.$$

The terms $f(a)$ and $-f(a)$ cancel each other. Also, the terms $-K_0 \cdot a$ and $K_0 \cdot a$ cancel each other. Thus, we get

$$f_L(x) - f_L(x') = K_0 \cdot x - K_0 \cdot x' = K_0 \cdot (x - x').$$

So, for the absolute value of the difference, taking into account that the absolute value of the product is equal to the product of absolute values, we get

$$|f_L(x) - f_L(x')| = |K_0| \cdot |x - x'|.$$

Here,

$$|K_0| = \left| \frac{f(b) - f(a)}{b - a} \right|.$$

The absolute value of the ratio is equal to the ratio of absolute values, so we have

$$|K_0| = \frac{|f(b) - f(a)|}{|b - a|},$$

i.e., $|K_0| = r$. Thus, for the function $f_L(x)$ obtained by linear interpolation, we have

$$|f_L(x) - f_L(x')| \le r \cdot |x - x'|.$$

In other words, for this function, the desired inequality is satisfied for the smallest possible value $K = r$.

So, linear interpolation is the most individually robust procedure. A natural question is: is linear interpolation the only one with this property? The answer is "yes", let us prove it.

Linear interpolation is the only most individually robust procedure: a proof. Let $f(x)$ be the most individually robust function, i.e., a function for which we have $|f(x) - f(x')| \le r \cdot |x - x'|$, where $r = \dfrac{|f(b) - f(a)|}{|b - a|}$. Let us prove that in this case, this function $f(x)$ comes from linear interpolation, i.e., for all x from the interval (a, b), we have

$$f(x) = f_L(x) = f(a) + \frac{f(b) - f(a)}{b - a} \cdot (x - a).$$

In principle, we can have two cases: when $f(a) < f(b)$ and when $f(b) < f(a)$. Let us consider the first case $f(a) < f(b)$.

Let us take any value $x \in (a, b)$. Linear interpolation always produces values in between $f(a)$ and $f(b)$, so in this case, $f(a) < f_L(x) < f(b)$.

1. Let us first prove that in this case, we cannot have

$$f(x) > f_L(x) = f(a) + \frac{f(b) - f(a)}{b - a} \cdot (x - a).$$

Indeed, by subtracting $f(a)$ from both sides of this inequality, we conclude that

$$f(x) - f(a) > \frac{f(b) - f(a)}{b - a} \cdot (x - a).$$

Here:

- We have $f(x) > f_L(x)$ and $f_L(x) > f(a)$, so $f(x) > f(a)$. Thus,

  $$f(x) - f(a) > 0, \text{ and } |f(x) - f(a)| = f(x) - f(a).$$

- We have $f(a) < f(b)$, so $f(b) - f(a) > 0$. Thus, $|f(b) - f(a)| = f(b) - f(a)$.
- We have $a < b$, so $b - a > 0$. Thus, $|b - a| = b - a$.
- Also, $a < x$, so $x - a > 0$. Thus, $|x - a| = x - a$.

So, the inequality

$$f(x) - f(a) > \frac{f(b) - f(a)}{b - a} \cdot (x - a)$$

implies that

$$|f(x) - f(a)| > \frac{|f(b) - f(a)|}{|b - a|} \cdot |x - a|.$$

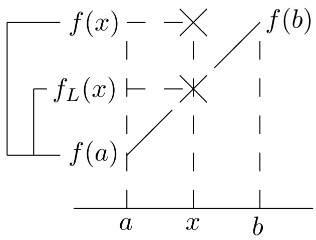
The ratio $\dfrac{|f(b) - f(a)|}{|b - a|}$ is what we denoted by r. Thus, we get

$$|f(x) - f(a)| > r \cdot |x - a|.$$

However, we assumed that the function $f(x)$ is maximally individually robust and therefore, that $|f(x) - f(x')| \le r \cdot |x - x'|$ for all x and x'. In particular, for $x' = a$, we conclude that

$$|f(x) - f(a)| \le r \cdot |x - a|,$$

which contradicts to the opposite inequality $|f(x) - f(a)| > r \cdot |x - a|$ that we derived. This contradictions shows that we cannot have $f(x) > f_L(x)$:



2. Let us now prove that in this case, we cannot have

$$f(x) < f_L(x) = f(a) + \frac{f(b) - f(a)}{b - a} \cdot (x - a).$$

Indeed, by subtracting, from $f(b)$, both sides of this inequality, we conclude that

$$f(b) - f(x) > (f(b) - f(a)) - \frac{f(b) - f(a)}{b - a} \cdot (x - a) =$$

$$(f(b) - f(a)) \cdot \left(1 - \frac{x - a}{b - a}\right) = (f(b) - f(a)) \cdot \frac{b - a - (x - a)}{b - a} =$$

$$(f(b) - f(a)) \cdot \frac{b - x}{b - a} = \frac{f(b) - f(a)}{b - a} \cdot (b - x).$$

Here:

- We have $f(x) < f_L(x)$ and $f_L(x) < f(b)$, so $f(x) < f(b)$. Thus,

$$f(b) - f(x) > 0, \text{ and } |f(b) - f(x)| = f(b) - f(x).$$

- We have $f(a) < f(b)$, so $f(b) - f(a) > 0$. Thus, $|f(b) - f(a)| = f(b) - f(a)$.

- We have $a < b$, so $b - a > 0$. Thus, $|b - a| = b - a$.
- Also, $x < b$, so $b - x > 0$. Thus, $|b - x| = b - x$.

So, the inequality

$$f(b) - f(x) > \frac{f(b) - f(a)}{b - a} \cdot (b - x)$$

implies that

$$|f(b) - f(x)| > \frac{|f(b) - f(a)|}{|b - a|} \cdot |b - x|.$$

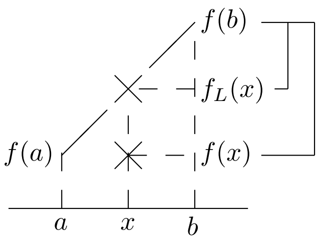
The ratio $\dfrac{|f(b) - f(a)|}{|b - a|}$ is what we denoted by r. Thus, we get

$$|f(b) - f(x)| > r \cdot |b - x|.$$

However, we assumed that the function $f(x)$ is maximally individually robust and therefore, that $|f(x') - f(x'')| \le r \cdot |x' - x''|$ for all x' and x''. In particular, for $x' = b$ and $x'' = x$, we conclude that

$$|f(b) - f(x)| \le r \cdot |b - x|,$$

which contradicts to the opposite inequality $|f(b) - f(x)| > r \cdot |b - x|$ that we derived. This contradictions shows that we cannot have $f(x) < f_L(x)$:



3. Since the value $f(x)$ cannot be larger than $f_L(x)$ and cannot be smaller that $f_L(x)$, it must be exactly equal to $f_L(x)$:

$$f(x) = f_L(x).$$

In other words, the function $f(x)$ must be obtained by linear interpolation.

Comment. In the case of $f(a) > f(b)$, the proof is similar.

9 The Most Individually Robust "And"-Operation

Formulation of the problem. Both inputs a and b to the "and"-operation are uncertain. So, it is reasonable to require that when we change both a and b by no more than some value ε – i.e., when the largest of these two changes does not exceed ε – then the value $f_{\&}(a,b)$ will change by no more than some constant K times this ε. In other words, we require that for all possible values a, a', b, and b', we have

$$|f_{\&}(a,b) - f_{\&}(a',b')| \le K \cdot \max(|a-a'|, |b-b'|).$$

We want to select an "and"-operation for which the corresponding coefficient K is the smallest possible.

A natural lower bound on K. For $a = b = 0$ and $a' = b' = 1$, we have $f_{\&}(0,0) = 0$, $f_{\&}(1,1) = 1$, and $\max(|a-a'|, |b-b'|) = 1$. So, for these values, the desired inequality takes the form $1 \le K$.

What happens for $\min(a,b)$. One can show that for $f_{\&}(a,b) = \min(a,b)$, the desired inequality is satisfied for $K = 1$ – and we know that this value is the smallest possible. So, we know that the min t-norm is the most individually robust.

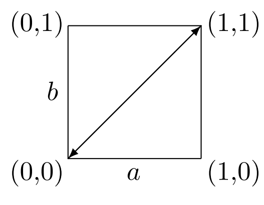
Let us prove that this the only most individually robust "and"-operation.

Min "and"-operation is the only most individually robust one: a proof. Suppose that have an "and"-operation which is the most individually robust, This means that for this operation, the desired inequality is satisfied for the smallest possible value K:

$$|f_{\&}(a,b) - f_{\&}(a',b')| \le \max(|a-a'|, |b-b'|).$$

In particular, for $a = b$ and $a' = b'$, we have

$$|f_{\&}(a,a) - f_{\&}(a',a')| \le |a-a'|.$$



In other words, for the function $F_0(a) = f_{\&}(a,a)$, we have

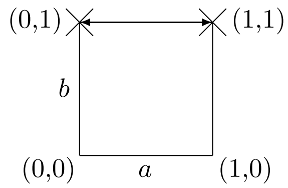
$$|F_0(a) - F_0(a')| \le |a-a'|.$$

For this function, we also have $F_0(0) = f_\&(0,0) = 0$ and $F_0(1) = f_\&(1,1) = 1$. In this case, $r = \dfrac{|F_0(1) - F_0(0)|}{|1-0|} = 1$. Thus, the function $F_0(a)$ satisfies the condition of individual robustness with the smallest possible coefficient K. We have already proven that in this case, this function is linear.

By applying linear interpolation to the values $F_0(0) = 0$ and $F_0(1) = 1$, we conclude that $F_0(a) = a$, i.e., that $f_\&(a,a) = a$.

Similarly, for $a' = b' = 1$, we have

$$|f_\&(a,1) - f_\&(a',1)| \le |a - a'|.$$



In other words, for a function $F_1(a) = f_\&(a,1)$, we have

$$|F_1(a) - F_1(a')| \le |a - a'|.$$

For this function, we also have $F_1(0) = f_\&(0,1) = 0$ and $F_1(1) = f_\&(1,1) = 1$. In this case, $r = \dfrac{|F_1(1) - F_1(0)|}{|1-0|} = 1$. Thus, the function $F_1(a)$ satisfies the condition of individual robustness with the smallest possible coefficient K. We have already proven that in this case, the function is linear.

By applying linear interpolation to the values $F_1(0) = 0$ and $F_1(1) = 1$, we conclude that $F_1(a) = a$, i.e., that $f_\&(a,1) = a$.

If $a \le b$, then, from $a \le b \le 1$ and monotonicity, we conclude that

$$f_\&(a,a) \le f_\&(a,b) \le f_\&(a,1).$$

We know that $f_\&(a,a) = a$ and $f_\&(a,1) = a$. Thus, we get $a \le f_\&(a,b) \le a$, hence $f_\&(a,b) = a$. Thus, when $a \le b$, we have $f_\&(a,b) = \min(a,b)$.

Due to commutativity, the same property is true for $a \ge b$: then

$$f_\&(a,b) = f_\&(b,a) = \min(a,b).$$

So, we always have $f_\&(a,b) = \min(a,b)$. Thus, the min "and"-operation is indeed the only most individually robust one.

10 Robustness vs. Individual Robustness: Example

Simplified situation: general description. To illustrate the difference between robustness and individual robustness, let us consider a simplified setting when we have only three possible value of the degree of confidence:

- the two "classical" values 0 ("false") and 1 ("true"), and
- the intermediate value 0.5 ("uncertain").

In this case, we have $3 \cdot 3 = 9$ possible pairs (a, b) of confidence degrees:

$(0.0, 1.0)$	$(0.5, 1.0)$	$(1.0, 1.0)$
$(0.0, 0.5)$	$(0.5, 0.5)$	$(1.0, 0.5)$
$(0.0, 0.0)$	$(0.5, 0.0)$	$(1.0, 0.0)$

What if we use algebraic product. If we use the algebraic product $f_\&(a, b) = a \cdot b$, we get the following values of $f_\&(a, b)$:

0.00	0.50	1.00
0.00	0.25	0.50
0.00	0.00	0.00

What is we use the minimum "and"-operation. If we use the minimum $f_\&(a, b) = \min(a, b)$, we get somewhat different values of $f_\&(a, b)$:

0.00	0.50	1.00
0.00	0.50	0.50
0.00	0.00	0.00

The only difference between these two “and”-operations is in the central value:

$$0.5 \cdot 0.5 = 0.25 \neq \min(0.5, 0.5) = 0.50.$$

Case of individual robustness. In individual robustness, we are interested in the largest possible difference between the values of the “and”-operation in the neighboring points.

Individual robustness: algebraic product. For the algebraic product, if we mark all neighboring points, we get the following picture:

$$\begin{tikzcd}
0.00 \arrow[r, leftrightarrow] \arrow[d, leftrightarrow] \arrow[dr, leftrightarrow] & 0.50 \arrow[r, leftrightarrow] \arrow[d, leftrightarrow] \arrow[dl, leftrightarrow] \arrow[dr, leftrightarrow] & 1.00 \arrow[d, leftrightarrow] \arrow[dl, leftrightarrow] \\
0.00 \arrow[r, leftrightarrow] \arrow[d, leftrightarrow] \arrow[dr, leftrightarrow] & 0.25 \arrow[r, leftrightarrow] \arrow[d, leftrightarrow] \arrow[dl, leftrightarrow] \arrow[dr, leftrightarrow] & 0.50 \arrow[d, leftrightarrow] \arrow[dl, leftrightarrow] \\
0.00 \arrow[r, leftrightarrow] & 0.00 \arrow[r, leftrightarrow] & 0.00
\end{tikzcd}$$

One can check that the largest difference between the values of the “and”-operation in the neighboring points is the difference:

$$f_{\&}(1,1) - f_{\&}(0,5,0.5) = 1 \cdot 1 - 0.5 \cdot 0.5 = 1 - 0.25 = 0.75.$$

Individual robustness: minimum “and”-operation. For the min operation, we can draw a similar picture:

$$\begin{tikzcd}
0.00 \arrow[r, leftrightarrow] \arrow[d, leftrightarrow] \arrow[dr, leftrightarrow] & 0.50 \arrow[r, leftrightarrow] \arrow[d, leftrightarrow] \arrow[dl, leftrightarrow] \arrow[dr, leftrightarrow] & 1.00 \arrow[d, leftrightarrow] \arrow[dl, leftrightarrow] \\
0.00 \arrow[r, leftrightarrow] \arrow[d, leftrightarrow] \arrow[dr, leftrightarrow] & \underline{0.50} \arrow[r, leftrightarrow] \arrow[d, leftrightarrow] \arrow[dl, leftrightarrow] \arrow[dr, leftrightarrow] & 0.50 \arrow[d, leftrightarrow] \arrow[dl, leftrightarrow] \\
0.00 \arrow[r, leftrightarrow] & 0.00 \arrow[r, leftrightarrow] & 0.00
\end{tikzcd}$$

To emphasize the difference between the two “and”-operations, we underlined the only different result. For the min operation, the largest possible difference is equal to

$$f_{\&}(1,1) - f_{\&}(0,5,0.5) = \min(1,1) - \min(0.5,0.5) = 1 - 0.5 = 0.5.$$

Individual robustness: conclusion. For the minimum "and"-operation, the largest difference (0.5) between the values of the "and"-operation in neighboring points is smaller than the corresponding value (0.75) for the algebraic product.

So, in this simplified setting – when we assume that we only have three possible degrees of certainty – the minimum "and"-operation is more individually robust than the algebraic product.

Robustness. Let us now compare these two "and"-operations from the viewpoint of robustness in general. In this case, we are interested:

- not in the largest difference,
- but in the sum of the squares of all the differences between the values of the "and"-operations at the neighboring points – which is exactly the squared difference between the corresponding tuples.

The only difference between these two situations is in the value of $f_\&(0.5, 0.5)$. So, to compare which sum is larger, its is sufficient to consider only the sums of the differences that involve this changing value – all other differences are the same for both "and"-operations.

Robustness: algebraic product. For the algebraic product, we get the following picture:

$$\begin{tikzcd}
0.00 \arrow[dr, leftrightarrow] & 0.50 \arrow[d, leftrightarrow] & 1.00 \arrow[dl, leftrightarrow] \\
0.00 \arrow[r, leftrightarrow] & 0.25 \arrow[r, leftrightarrow] \arrow[dl, leftrightarrow] \arrow[d, leftrightarrow] \arrow[dr, leftrightarrow] & 0.50 \\
0.00 & 0.00 & 0.00
\end{tikzcd}$$

In this case, we have seven differences equal to 0.25 and one difference equal to 0.75. So, the sum of the squares of these differences is equal to

$$7 \cdot 0.25^2 + 0.75^2 = 7 \cdot 0.0625 + 0.5625 = 0.4375 + 0.5625 = 1.0$$

Robustness: minimum "and"-operation. For the minimum "and"-operation, we get the following picture:

$$\begin{tikzcd}
0.00 & 0.50 & 1.00 \\
0.00 & 0.50 \arrow[ul, leftrightarrow] \arrow[u, leftrightarrow] \arrow[ur, leftrightarrow] \arrow[l, leftrightarrow] \arrow[r, leftrightarrow] \arrow[dl, leftrightarrow] \arrow[d, leftrightarrow] \arrow[dr, leftrightarrow] & 0.50 \\
0.00 & 0.00 & 0.00
\end{tikzcd}$$

In this case, we have 6 differences equal to 0.5 and 2 differences equal to 0. So, the sum of the squares of these differences is equal to

$$6 \cdot 0.5^2 + 2 \cdot 0^2 = 6 \cdot 0.25 = 1.5.$$

Robustness: conclusion. The sum 1.5 corresponding to the minimum "and"-operation is larger than sum 1.0 corresponding to the algebraic product.

Thus, in this simplified setting, the algebraic product is more robust than the minimum "and"-operation.

Summarizing. In this section, we consider a simplified setting, when we have only three degrees of certainty: 0, 1, and 0.5. In this setting, if we use algebraic product, then:

- on average, the differences between the values of the "and"-operation in the neighboring points are smaller than for the minimum, but
- it is possible that this difference becomes equal to 0.75 – which is larger than the largest difference 0.5 corresponding to the minimum "and"-operation.

In other words:

- On the one hand, in this setting, algebraic product is more *robust* than the minimum. Indeed, for the algebraic product, the squared distance between the corresponding tuples – which is equal to the sum of the differences between the values of the "and"-operation in neighboring points – is smaller.
- On the other hand, in this setting, minimum is more *individually robust* than the algebraic product. Indeed for the minimum "and"-operation, the largest difference between the values of the "and"-operation in neighboring points – which is a measure of individual robustness – is smaller.

11 The Most Individually Robust "Or"-Operation

Formulation of the problem. Let us now go back from the simplified setting to the original setting, when all possible values from the interval $[0,1]$ are possible values of degree of certainty.

Both inputs a and b to the "or"-operation are uncertain. So, it is reasonable to require that when we change both a and b by no more than some value ε, then the value $f_\vee(a,b)$ will change by no more than some constant K times this ε. In other words, we require that for all possible values a, a', b, and b', we have

$$|f_\vee(a,b) - f_\vee(a',b')| \le K \cdot \max(|a-a'|, |b-b'|).$$

We want to select an "or"-operation for which the corresponding coefficient K is the smallest possible.

A natural lower bound on K. For $a = b = 0$ and $a' = b' = 1$, we have $f_\vee(0,0) = 0$, $f_\vee(1,1) = 1$, and $\max(|a-a'|, |b-b'|) = 1$. So, for these values, the desired inequality takes the form $1 \le K$.

What happens for $\max(a,b)$. One can show that for $f_\vee(a,b) = \max(a,b)$, the desired inequality is satisfied for $K = 1$ – and we know that this value is the smallest possible. So, we know that the max t-norm is the most individually robust.

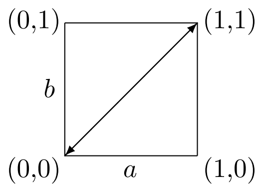
Let us prove that this the only most individually robust "or"-operation.

Max "or"-operation is the only most individually robust one: a proof. Suppose that have an "or"-operation which is the most individually robust, This means that for this operation, the desired inequality is satisfied for the smallest possible value K:

$$|f_\vee(a,b) - f_\vee(a',b')| \le \max(|a-a'|, |b-b'|).$$

In particular, for $a = b$ and $a' = b'$, we have

$$|f_\vee(a,a) - f_\vee(a',a')| \le |a-a'|.$$



In other words, for the function $G_0(a) = f_\vee(a,a)$, we have

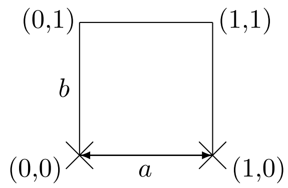
$$|G_0(a) - G_0(a')| \le |a-a'|.$$

For this function, we also have $G_0(0) = f_\vee(0,0) = 0$ and $G_0(1) = f_\vee(1,1) = 1$. In this case, $r = \dfrac{|G_0(1) - G_0(0)|}{|1-0|} = 1$. Thus, the function $G_0(a)$ satisfies the condition of individual robustness with the smallest possible coefficient K. We have already proven that in this case, the function is linear.

By applying linear interpolation to the values $G_0(0) = 0$ and $G_0(1) = 1$, we conclude that $G_0(a) = a$, i.e., that $f_\vee(a,a) = a$.

Similarly, for $a' = b' = 0$, we have

$$|f_\vee(a,0) - f_\vee(a',0)| \le |a - a'|.$$



In other words, for a function $G_1(a) = f_\vee(a,0)$, we have

$$|G_1(a) - G_1(a')| \le |a - a'|.$$

For this function, we also have $G_1(0) = f_\vee(0,0) = 0$ and $G_1(1) = f_\vee(1,0) = 1$. In this case, $r = \dfrac{|G_1(1) - G_1(0)|}{|1-0|} = 1$. Thus, the function $G_1(a)$ satisfies the condition of individual robustness with the smallest possible coefficient K. We have already proven that in this case, the function is linear.

By applying linear interpolation to the values $G_1(0) = 0$ and $G_1(1) = 1$, we conclude that $G_1(a) = a$, i.e., that $f_\vee(a,0) = a$.

If $b \le a$, then, from $0 \le b \le a$ and monotonicity, we conclude that

$$f_\vee(a,0) \le f_\vee(a,b) \le f_\vee(a,a).$$

We know that $f_\vee(a,0) = a$ and $f_\vee(a,a) = a$. Thus, we get $a \le f_\vee(a,b) \le a$, hence $f_\vee(a,b) = a$. Thus, when $b \le a$, we have $f_\vee(a,b) = \max(a,b)$.

Due to commutativity, the same property is true for $a \le b$: then

$$f_\vee(a,b) = f_\vee(b,a) = \max(a,b).$$

So, we always have $f_\vee(a,b) = \max(a,b)$. Thus, the max "or"-operation is indeed the most individually robust one.

12 General Conclusion

- If we are controlling a group of objects, and we want to achieve the best overall result, we should use

$$f_{\&}(a,b) = a \cdot b \text{ and } f_{\vee}(a,b) = a + b - a \cdot b.$$

- If we are controlling an individual object, and we want to achieve the best result for this object, we should use

$$f_{\&}(a,b) = \min(a,b) \text{ and } f_{\vee}(a,b) = \max(a,b).$$