Pólya's Proof of the Weighted Arithmetic–Geometric Mean Inequality

Manuel Eberl

March 17, 2025

Abstract

This article provides a formalisation of the Weighted Arithmetic–Geometric Mean Inequality: given non-negative reals $a_1, \ldots, a_n$ and non-negative weights $w_1, \ldots, w_n$ such that $w_1 + \ldots + w_n = 1$, we have

$$\prod_{i=1}^{n} a_i^{w_i} \leq \sum_{i=1}^{n} w_i a_i \ .$$

If the weights are additionally all non-zero, equality holds if and only if $a_1 = \ldots = a_n$.

As a corollary with $w_1 = \ldots = w_n = \frac{1}{n}$, the regular arithmetic–geometric mean inequality follows, namely that

$$\sqrt[n]{a_1 \ldots a_n} \leq \frac{1}{n}(a_1 + \ldots + a_n) \ .$$

I follow Pólya's elegant proof, which uses the inequality $1 + x \leq e^x$ as a starting point. Pólya claims that this proof came to him in a dream, and that it was 'the best mathematics he had ever dreamt." [1, pp. 22–26]

Contents

1 The Weighted Arithmetic–Geometric Mean Inequality

theory *Weighted-Arithmetic-Geometric-Mean*
  imports *Complex-Main*
begin

1.1 Auxiliary Facts

lemma *root-powr-inverse'*: $0 < n \Longrightarrow 0 \leq x \Longrightarrow \mathit{root}\ n\ x = x\ \mathit{powr}\ (1/n)$
  by (*cases* $x = 0$) (*auto simp*: *root-powr-inverse*)

lemma *powr-sum-distrib-real-right*:
  assumes $a \neq 0$
  shows $(\prod x{\in}X.\ a\ \mathit{powr}\ e\ x :: \mathit{real}) = a\ \mathit{powr}\ (\sum x{\in}X.\ e\ x)$
  using *assms*
  by (*induction X rule*: *infinite-finite-induct*) (*auto simp*: *powr-add*)

lemma *powr-sum-distrib-real-left*:
  assumes $\bigwedge x.\ x \in X \Longrightarrow a\ x \geq 0$
  shows $(\prod x{\in}X.\ a\ x\ \mathit{powr}\ e :: \mathit{real}) = (\prod x{\in}X.\ a\ x)\ \mathit{powr}\ e$
  using *assms*
  by (*induction X rule*: *infinite-finite-induct*)
    (*auto simp*: *powr-mult prod-nonneg*)

lemma *prod-ge-pointwise-le-imp-pointwise-eq*:
  fixes $f :: 'a \Rightarrow \mathit{real}$
  assumes *finite X*
  assumes *ge*: $\mathit{prod}\ f\ X \geq \mathit{prod}\ g\ X$
  assumes *nonneg*: $\bigwedge x.\ x \in X \Longrightarrow f\ x \geq 0$
  assumes *pos*: $\bigwedge x.\ x \in X \Longrightarrow g\ x > 0$
  assumes *le*: $\bigwedge x.\ x \in X \Longrightarrow f\ x \leq g\ x$ **and** *x*: $x \in X$
  shows $f\ x = g\ x$
proof (*rule ccontr*)
  assume $f\ x \neq g\ x$
  with *le*[*of x*] **and** *x* **have** $f\ x < g\ x$
    by *auto*
  hence $\mathit{prod}\ f\ X < \mathit{prod}\ g\ X$
    using *x* **and** *le* **and** *nonneg* **and** *pos* **and** ‹*finite X*›
    by (*intro prod-mono-strict*) *auto*
  with *ge* **show** *False*
    by *simp*
qed

lemma *powr-right-real-eq-iff*:
  assumes $a \geq (0 :: \mathit{real})$
  shows $a\ \mathit{powr}\ x = a\ \mathit{powr}\ y \longleftrightarrow a = 0 \vee a = 1 \vee x = y$
  using *assms* **by** (*auto simp*: *powr-def*)

```
lemma powr-left-real-eq-iff:
  assumes a ≥ (0 :: real) b ≥ 0 x ≠ 0
  shows   a powr x = b powr x ⟷ a = b
  using assms by (auto simp: powr-def)

lemma exp-real-eq-one-plus-iff:
  fixes x :: real
  shows exp x = 1 + x ⟷ x = 0
proof (cases x = 0)
  case False
  define f :: real ⇒ real where f = (λx. exp x − 1 − x)
  have deriv: (f has-field-derivative (exp x − 1)) (at x) for x
    by (auto simp: f-def intro!: derivative-eq-intros)

  have ∃z. z>min x 0 ∧ z < max x 0 ∧ f (max x 0) − f (min x 0) =
           (max x 0 − min x 0) ∗ (exp z − 1)
    using MVT2[of min x 0 max x 0 f λx. exp x − 1] deriv False
    by (auto simp: min-def max-def)
  then obtain z where z ∈ {min x 0<..<max x 0}
    f (max x 0) − f (min x 0) = (max x 0 − min x 0) ∗ (exp z − 1)
    by (auto simp: f-def)
  thus ?thesis using False
    by (cases x 0 :: real rule: linorder-cases) (auto simp: f-def)
qed auto
```

1.2 The Inequality

We first prove the equality under the assumption that all the a_i and w_i are positive.

```
lemma weighted-arithmetic-geometric-mean-pos:
  fixes a w :: 'a ⇒ real
  assumes finite X
  assumes pos1: ⋀x. x ∈ X ⟹ a x > 0
  assumes pos2: ⋀x. x ∈ X ⟹ w x > 0
  assumes sum-weights: (∑x∈X. w x) = 1
  shows   (∏x∈X. a x powr w x) ≤ (∑x∈X. w x ∗ a x)
proof −
  note nonneg1 = less-imp-le[OF pos1]
  note nonneg2 = less-imp-le[OF pos2]
  define A where A = (∑x∈X. w x ∗ a x)
  define r where r = (λx. a x / A − 1)
  from sum-weights have X ≠ {} by auto
  hence A ≠ 0
    unfolding A-def using nonneg1 nonneg2 pos1 pos2 ‹finite X›
    by (subst sum-nonneg-eq-0-iff) force+
  moreover from nonneg1 nonneg2 have A ≥ 0
    by (auto simp: A-def intro!: sum-nonneg)
  ultimately have A > 0 by simp
```

```
  have (∏x∈X. (1 + r x) powr w x) = (∏x∈X. (a x / A) powr w x)
    by (simp add: r-def)
  also have ... = (∏x∈X. a x powr w x) / (∏x∈X. A powr w x)
    unfolding prod-dividef [symmetric]
      using assms pos2 ‹A > 0› by (intro prod.cong powr-divide) (auto intro:
less-imp-le)
  also have (∏x∈X. A powr w x) = exp ((∑x∈X. w x) * ln A)
   using ‹A > 0› and ‹finite X› by (simp add: powr-def exp-sum sum-distrib-right)
  also have (∑x∈X. w x) = 1 by fact
  also have exp (1 * ln A) = A
    using ‹A > 0› by simp
  finally have lhs: (∏x∈X. (1 + r x) powr w x) = (∏x∈X. a x powr w x) / A .

  have (∏x∈X. exp (w x * r x)) = exp (∑x∈X. w x * r x)
    using ‹finite X› by (simp add: exp-sum)
  also have (∑x∈X. w x * r x) = (∑x∈X. a x * w x) / A − 1
   using ‹A > 0› by (simp add: r-def algebra-simps sum-subtractf sum-divide-distrib
sum-weights)
  also have (∑x∈X. a x * w x) / A = 1
    using ‹A > 0› by (simp add: A-def mult.commute)
  finally have rhs: (∏x∈X. exp (w x * r x)) = 1 by simp

  have (∏x∈X. a x powr w x) / A = (∏x∈X. (1 + r x) powr w x)
    by (fact lhs [symmetric])
  also have (∏x∈X. (1 + r x) powr w x) ≤ (∏x∈X. exp (w x * r x))
  proof (intro prod-mono conjI)
    fix x assume x: x ∈ X
    have 1 + r x ≤ exp (r x)
      by (rule exp-ge-add-one-self)
    hence (1 + r x) powr w x ≤ exp (r x) powr w x
      using nonneg1[of x] nonneg2[of x] x ‹A > 0›
      by (intro powr-mono2) (auto simp: r-def field-simps)
    also have ... = exp (w x * r x)
      by (simp add: powr-def)
    finally show (1 + r x) powr w x ≤ exp (w x * r x) .
  qed auto
  also have (∏x∈X. exp (w x * r x)) = 1 by (fact rhs)
  finally show (∏x∈X. a x powr w x) ≤ A
    using ‹A > 0› by (simp add: field-simps)
qed
```

We can now relax the positivity assumptions to non-negativity: if one of the a_i is zero, the theorem becomes trivial (note that $0^0 = 0$ by convention for the real-valued power operator (*powr*)).

Otherwise, we can simply remove all the indices that have weight 0 and apply the above auxiliary version of the theorem.

```
theorem weighted-arithmetic-geometric-mean:
  fixes a w :: 'a ⇒ real
  assumes finite X
```

```
  assumes nonneg1: ⋀x. x ∈ X ⟹ a x ≥ 0
  assumes nonneg2: ⋀x. x ∈ X ⟹ w x ≥ 0
  assumes sum-weights: (∑x∈X. w x) = 1
  shows   (∏x∈X. a x powr w x) ≤ (∑x∈X. w x ∗ a x)
proof (cases ∃x∈X. a x = 0)
  case True
  hence (∏x∈X. a x powr w x) = 0
    using ‹finite X› by simp
  also have … ≤ (∑x∈X. w x ∗ a x)
    by (intro sum-nonneg mult-nonneg-nonneg assms)
  finally show ?thesis .
next
  case False
  have (∑x∈X−{x. w x = 0}. w x) = (∑x∈X. w x)
    by (intro sum.mono-neutral-left assms) auto
  also have … = 1 by fact
  finally have sum-weights′: (∑x∈X−{x. w x = 0}. w x) = 1 .

  have (∏x∈X. a x powr w x) = (∏x∈X−{x. w x = 0}. a x powr w x)
    using ‹finite X› False by (intro prod.mono-neutral-right) auto
  also have … ≤ (∑x∈X−{x. w x = 0}. w x ∗ a x) using assms False
    by (intro weighted-arithmetic-geometric-mean-pos sum-weights′)
       (auto simp: order.strict-iff-order)
  also have … = (∑x∈X. w x ∗ a x)
    using ‹finite X› by (intro sum.mono-neutral-left) auto
  finally show ?thesis .
qed
```

We can derive the regular arithmetic/geometric mean inequality from this by simply setting all the weights to $\frac{1}{n}$:

```
corollary arithmetic-geometric-mean:
  fixes a :: ′a ⇒ real
  assumes finite X
  defines n ≡ card X
  assumes nonneg: ⋀x. x ∈ X ⟹ a x ≥ 0
  shows   root n (∏x∈X. a x) ≤ (∑x∈X. a x) / n
proof (cases X = {})
  case False
  with assms have n: n > 0
    by auto
  have (∏x∈X. a x powr (1 / n)) ≤ (∑x∈X. (1 / n) ∗ a x)
    using n assms by (intro weighted-arithmetic-geometric-mean) auto
  also have (∏x∈X. a x powr (1 / n)) = (∏x∈X. a x) powr (1 / n)
    using nonneg by (subst powr-sum-distrib-real-left) auto
  also have … = root n (∏x∈X. a x)
    using ‹n > 0› nonneg by (subst root-powr-inverse′) (auto simp: prod-nonneg)
  also have (∑x∈X. (1 / n) ∗ a x) = (∑x∈X. a x) / n
    by (subst sum-distrib-left [symmetric]) auto
  finally show ?thesis .
```

qed (*auto simp*: *n-def*)

1.3 The Equality Case

Next, we show that weighted arithmetic and geometric mean are equal if and only if all the a_i are equal.

We first prove the more difficult direction as a lemmas and again first assume positivity of all a_i and w_i and will relax this somewhat later.

lemma *weighted-arithmetic-geometric-mean-eq-iff-pos*:
  fixes *a w* :: *'a ⇒ real*
  assumes *finite X*
  assumes *pos1*: ⋀*x*. *x* ∈ *X* ⟹ *a x* > *0*
  assumes *pos2*: ⋀*x*. *x* ∈ *X* ⟹ *w x* > *0*
  assumes *sum-weights*: (∑*x*∈*X*. *w x*) = *1*
  assumes *eq*: (∏*x*∈*X*. *a x powr w x*) = (∑*x*∈*X*. *w x* ∗ *a x*)
  shows ∀*x*∈*X*. ∀*y*∈*X*. *a x* = *a y*
proof −
  note *nonneg1* = *less-imp-le*[*OF pos1*]
  note *nonneg2* = *less-imp-le*[*OF pos2*]
  define *A* **where** *A* = (∑*x*∈*X*. *w x* ∗ *a x*)
  define *r* **where** *r* = (λ*x*. *a x* / *A* − *1*)
  from *sum-weights* **have** *X* ≠ {} **by** *auto*
  hence *A* ≠ *0*
    unfolding *A-def* **using** *nonneg1 nonneg2 pos1 pos2* ‹*finite X*›
    by (*subst sum-nonneg-eq-0-iff*) *force+*
  moreover from *nonneg1 nonneg2* **have** *A* ≥ *0*
    by (*auto simp*: *A-def intro*!: *sum-nonneg*)
  ultimately have *A* > *0* **by** *simp*

  have *r-ge*: *r x* ≥ −*1* **if** *x*: *x* ∈ *X* **for** *x*
    using ‹*A* > *0*› *pos1*[*OF x*] **by** (*auto simp*: *r-def field-simps*)

  have (∏*x*∈*X*. (*1* + *r x*) *powr w x*) = (∏*x*∈*X*. (*a x* / *A*) *powr w x*)
    by (*simp add*: *r-def*)
  also have ... = (∏*x*∈*X*. *a x powr w x*) / (∏*x*∈*X*. *A powr w x*)
    unfolding *prod-dividef* [*symmetric*]
      using *assms pos2* ‹*A* > *0*› **by** (*intro prod.cong powr-divide*) (*auto intro*: *less-imp-le*)
  also have (∏*x*∈*X*. *A powr w x*) = *exp* ((∑*x*∈*X*. *w x*) ∗ *ln A*)
   using ‹*A* > *0*› **and** ‹*finite X*› **by** (*simp add*: *powr-def exp-sum sum-distrib-right*)
  also have (∑*x*∈*X*. *w x*) = *1* **by** *fact*
  also have *exp* (*1* ∗ *ln A*) = *A*
    using ‹*A* > *0*› **by** *simp*
  finally have *lhs*: (∏*x*∈*X*. (*1* + *r x*) *powr w x*) = (∏*x*∈*X*. *a x powr w x*) / *A* **.**

  have (∏*x*∈*X*. *exp* (*w x* ∗ *r x*)) = *exp* (∑*x*∈*X*. *w x* ∗ *r x*)
    using ‹*finite X*› **by** (*simp add*: *exp-sum*)
  also have (∑*x*∈*X*. *w x* ∗ *r x*) = (∑*x*∈*X*. *a x* ∗ *w x*) / *A* − *1*

```
  using ‹A > 0› by (simp add: r-def algebra-simps sum-subtractf sum-divide-distrib
sum-weights)
  also have (∑x∈X. a x ∗ w x) / A = 1
    using ‹A > 0› by (simp add: A-def mult.commute)
  finally have rhs: (∏x∈X. exp (w x ∗ r x)) = 1 by simp

  have a x = A if x: x ∈ X for x
  proof −
    have (1 + r x) powr w x = exp (w x ∗ r x)
    proof (rule prod-ge-pointwise-le-imp-pointwise-eq
            [where f = λx. (1 + r x) powr w x and g = λx. exp (w x ∗ r x)])
      show (1 + r x) powr w x ≤ exp (w x ∗ r x) if x: x ∈ X for x
      proof −
        have 1 + r x ≤ exp (r x)
          by (rule exp-ge-add-one-self)
        hence (1 + r x) powr w x ≤ exp (r x) powr w x
          using nonneg1[of x] nonneg2[of x] x ‹A > 0›
          by (intro powr-mono2) (auto simp: r-def field-simps)
        also have ... = exp (w x ∗ r x)
          by (simp add: powr-def)
        finally show (1 + r x) powr w x ≤ exp (w x ∗ r x) .
      qed
    next
      show (∏x∈X. (1 + r x) powr w x) ≥ (∏x∈X. exp (w x ∗ r x))
      proof −
        have (∏x∈X. (1 + r x) powr w x) = (∏x∈X. a x powr w x) / A
          by (fact lhs)
        also have ... = 1
          using ‹A ≠ 0› by (simp add: eq A-def)
        also have ... = (∏x∈X. exp (w x ∗ r x))
          by (simp add: rhs)
        finally show ?thesis by simp
      qed
    qed (use x ‹finite X› in auto)

    also have exp (w x ∗ r x) = exp (r x) powr w x
      by (simp add: powr-def)
    finally have 1 + r x = exp (r x)
      using x pos2[of x] r-ge[of x] by (subst (asm) powr-left-real-eq-iff) auto
    hence r x = 0
      using exp-real-eq-one-plus-iff[of r x] by auto
    hence a x = A
      using ‹A > 0› by (simp add: r-def field-simps)
    thus ?thesis
      by (simp add: )
  qed
  thus ∀x∈X. ∀y∈X. a x = a y
    by auto
qed
```

We can now show the full theorem and relax the positivity condition on the a_i to non-negativity. This is possible because if some a_i is zero and the two means coincide, then the product is obviously 0, but the sum can only be 0 if *all* the a_i are 0.

```
theorem weighted-arithmetic-geometric-mean-eq-iff:
  fixes a w :: 'a ⇒ real
  assumes finite X
  assumes nonneg1: ⋀x. x ∈ X ⟹ a x ≥ 0
  assumes pos2:    ⋀x. x ∈ X ⟹ w x > 0
  assumes sum-weights: (∑x∈X. w x) = 1
  shows   (∏x∈X. a x powr w x) = (∑x∈X. w x ∗ a x) ⟷ X ≠ {} ∧ (∀x∈X.
∀y∈X. a x = a y)
proof
  assume ∗: X ≠ {} ∧ (∀x∈X. ∀y∈X. a x = a y)
  from ∗ have X ≠ {}
    by blast

  from ∗ obtain c where c: ⋀x. x ∈ X ⟹ a x = c c ≥ 0
  proof (cases X = {})
    case False
    then obtain x where x ∈ X by blast
    thus ?thesis using ∗ that[of a x] nonneg1[of x] by metis
  next
    case True
    thus ?thesis
      using that[of 1] by auto
  qed

  have (∏x∈X. a x powr w x) = (∏x∈X. c powr w x)
    by (simp add: c)
  also have ... = c
   using assms c ‹X ≠ {}› by (cases c = 0) (auto simp: powr-sum-distrib-real-right)
  also have ... = (∑x∈X. w x ∗ a x)
    using sum-weights by (simp add: c(1) flip: sum-distrib-left sum-distrib-right)
  finally show (∏x∈X. a x powr w x) = (∑x∈X. w x ∗ a x) .
next
  assume ∗: (∏x∈X. a x powr w x) = (∑x∈X. w x ∗ a x)
  have X ≠ {}
    using ∗ by auto
  moreover have (∀x∈X. ∀y∈X. a x = a y)
  proof (cases ∃x∈X. a x = 0)
    case False
    with nonneg1 have pos1: ∀x∈X. a x > 0
      by force
    thus ?thesis
      using weighted-arithmetic-geometric-mean-eq-iff-pos[of X a w] assms ∗
      by blast
  next
    case True
```

hence $(\prod x \in X.\ a\ x\ powr\ w\ x) = 0$
using *assms* **by** *auto*
with $*$ **have** $(\sum x \in X.\ w\ x * a\ x) = 0$
by *auto*
also have *?this* $\longleftrightarrow (\forall x \in X.\ w\ x * a\ x = 0)$
using *assms* **by** (*intro sum-nonneg-eq-0-iff mult-nonneg-nonneg*) (*auto intro*: *less-imp-le*)
finally have $(\forall x \in X.\ a\ x = 0)$
using *pos2* **by** *force*
thus *?thesis*
by *auto*
qed
ultimately show $X \neq \{\} \wedge (\forall x \in X.\ \forall y \in X.\ a\ x = a\ y)$
by *blast*
qed

Again, we derive a version for the unweighted arithmetic/geometric mean.

corollary *arithmetic-geometric-mean-eq-iff*:
fixes $a :: 'a \Rightarrow real$
assumes *finite X*
defines $n \equiv card\ X$
assumes *nonneg*: $\bigwedge x.\ x \in X \Longrightarrow a\ x \geq 0$
shows $root\ n\ (\prod x \in X.\ a\ x) = (\sum x \in X.\ a\ x)\ /\ n \longleftrightarrow (\forall x \in X.\ \forall y \in X.\ a\ x = a\ y)$
proof (*cases* $X = \{\}$)
case *False*
with *assms* **have** $n > 0$
by *auto*
have $(\prod x \in X.\ a\ x\ powr\ (1\ /\ n)) = (\sum x \in X.\ (1\ /\ n) * a\ x) \longleftrightarrow$
$X \neq \{\} \wedge (\forall x \in X.\ \forall y \in X.\ a\ x = a\ y)$
using *assms False* **by** (*intro weighted-arithmetic-geometric-mean-eq-iff*) *auto*
also have $(\prod x \in X.\ a\ x\ powr\ (1\ /\ n)) = (\prod x \in X.\ a\ x)\ powr\ (1\ /\ n)$
using *nonneg* **by** (*subst powr-sum-distrib-real-left*) *auto*
also have $\ldots = root\ n\ (\prod x \in X.\ a\ x)$
using ‹$n > 0$› *nonneg* **by** (*subst root-powr-inverse'*) (*auto simp*: *prod-nonneg*)
also have $(\sum x \in X.\ (1\ /\ n) * a\ x) = (\sum x \in X.\ a\ x)\ /\ n$
by (*subst sum-distrib-left* [*symmetric*]) *auto*
finally show *?thesis* **using** *False* **by** *auto*
qed (*auto simp*: *n-def*)

1.4 The Binary Version

For convenience, we also derive versions for only two numbers:

corollary *weighted-arithmetic-geometric-mean-binary*:
fixes *w1 w2 x1 x2* :: *real*
assumes $x1 \geq 0$ $x2 \geq 0$ $w1 \geq 0$ $w2 \geq 0$ $w1 + w2 = 1$
shows $x1\ powr\ w1 * x2\ powr\ w2 \leq w1 * x1 + w2 * x2$
proof $-$
let $?a = \lambda b.\ if\ b\ then\ x1\ else\ x2$

let *?w* = λb. *if b then w1 else w2*
from *assms* **have** ($\prod x \in UNIV$. *?a x powr ?w x*) $\le$ ($\sum x \in UNIV$. *?w x* $*$ *?a x*)
by (*intro weighted-arithmetic-geometric-mean*) (*auto simp add*: *UNIV-bool*)
thus *?thesis* **by** (*simp add*: *UNIV-bool add-ac mult-ac*)
qed

corollary *weighted-arithmetic-geometric-mean-eq-iff-binary*:
fixes *w1 w2 x1 x2* :: *real*
assumes *x1* $\ge$ *0 x2* $\ge$ *0 w1* > *0 w2* > *0 w1* + *w2* = *1*
shows *x1 powr w1* $*$ *x2 powr w2* = *w1* $*$ *x1* + *w2* $*$ *x2* $\longleftrightarrow$ *x1* = *x2*
proof −
let *?a* = λb. *if b then x1 else x2*
let *?w* = λb. *if b then w1 else w2*
from *assms* **have** ($\prod x \in UNIV$. *?a x powr ?w x*) = ($\sum x \in UNIV$. *?w x* $*$ *?a x*)
$\longleftrightarrow$ (*UNIV* :: *bool set*) $\neq$ {} $\wedge$ ($\forall x \in UNIV$. $\forall y \in UNIV$. *?a x* =
?a y)
by (*intro weighted-arithmetic-geometric-mean-eq-iff*) (*auto simp add*: *UNIV-bool*)
thus *?thesis* **by** (*auto simp*: *UNIV-bool add-ac mult-ac*)
qed

corollary *arithmetic-geometric-mean-binary*:
fixes *x1 x2* :: *real*
assumes *x1* $\ge$ *0 x2* $\ge$ *0*
shows *sqrt* (*x1* $*$ *x2*) $\le$ (*x1* + *x2*) / *2*
using *weighted-arithmetic-geometric-mean-binary*[*of x1 x2 1/2 1/2*] *assms*
by (*simp add*: *powr-half-sqrt field-simps real-sqrt-mult*)

corollary *arithmetic-geometric-mean-eq-iff-binary*:
fixes *x1 x2* :: *real*
assumes *x1* $\ge$ *0 x2* $\ge$ *0*
shows *sqrt* (*x1* $*$ *x2*) = (*x1* + *x2*) / *2* $\longleftrightarrow$ *x1* = *x2*
using *weighted-arithmetic-geometric-mean-eq-iff-binary*[*of x1 x2 1/2 1/2*] *assms*
by (*simp add*: *powr-half-sqrt field-simps real-sqrt-mult*)

end

References

[1] J. M. Steele. *The Cauchy–Schwarz Master Class: An Introduction to the Art of Mathematical Inequalities.* Cambridge University Press, 2004.