Writing mathematics in sentences

Last modified: 2026-11-07 21:34:25 (PST)

> **AI-Assisted Authorship**
>
> This chapter was written with the assistance of GitHub Copilot, which expanded on outlined ideas and draft notes provided by the author. The content represents the author's perspective and has been reviewed for accuracy, but the detailed prose was generated through AI assistance.

A mathematical paper is made of sentences, and the formulas are part of the sentences. This chapter collects rules for fitting formulas into prose. It draws on the lecture notes of a Stanford course on mathematical writing (Knuth et al. 1987). This chapter selects the rules that bear most on formulas. Several of the course's other rules appear in other chapters, and Table 1 points to them. The notes also cover spelling, punctuation and refereeing, which this book does not.

Keep formulas inside sentences

Readers encounter a formula as part of a sentence, so the words around it decide whether the formula can be read.

Separate formulas with words

Put words between two formulas. Without them, the reader cannot tell where one formula ends and the next begins.

Example 0.1 (Separating two formulas)**.**

✗ Define f_k, $k \geq 2$.

✓ Define f_k, where $k \geq 2$.

In the ✗ version the comma between the two formulas seems to belong to the first one. The ✓ version names the relation between them.

Do not start a sentence with a symbol

A sentence that starts with a symbol is hard to see as a new sentence, because the formula runs on from the period before it. Start with a word, such as the name of the object.

Example 0.2 (Starting a sentence with a word)**.**

✗ $t^2 - 5$ has two real roots.

✓ The polynomial $t^2 - 5$ has two real roots.

Write logical symbols as words

In prose, write "implies", "for every" and "there exists", not $\Rightarrow$, $\forall$ and $\exists$. The same goes for the symbols for "therefore" and "such that". Keep the symbols in a work on logic.

Test the sentence with the formulas removed

Many readers skim past the formulas on a first reading. Replace each formula by a placeholder such as "blah". If the sentence no longer reads smoothly, rewrite it.

Tie formulas together with commentary

A list of formulas with no words between them is the style of a homework paper. Say what each step does and why the next one follows. A sentence such as "Because the set S has an element, we can pick one and call it s" tells the reader where the argument is going, and the bare formula does not.

Make the prose say what the formula says

The sentence that introduces or interprets a formula must describe that formula. Check every term in the sentence against the formula, symbol by symbol. The words and the symbols must name the same quantity, the same data, and the same kind of average. When you change one, change the other in the same edit.

Example 0.3 (A sentence that names the wrong average)**.** Suppose the formula is $\mathbb{E}[(Y_* - \hat{y}(x))^2]$, a population mean at one covariate value x. The sentence "the average squared error over the training set" does not describe it. It names a sample average over many observations. Write "the expected squared prediction error at x" instead.

Before you submit, reread each formula next to the sentence that describes it. Do the same after every edit to either one. Related rules cover using one notation for one thing and defining every variable when you introduce it.

Write each sentence in words or in math, not both

A sentence is either entirely in words or entirely in math. A single inline expression inside a word sentence is fine, as in "the prediction function f equals the conditional mean". A sentence that states one condition in words and a second condition as a relation between expressions is a hybrid. Split it into two complete sentences: the first states the result in words, and the second states it as a displayed formula.

Example 0.4 (Splitting a hybrid sentence)**.**

✗ The reducible risk is zero if and only if $\mathbb{P}(f(X) \neq \mathbb{E}[Y \mid X]) = 0$.

✓ The reducible risk is zero if and only if the prediction function differs from the conditional mean with probability 0.

$$\begin{aligned} &\mathbb{E}\left[(\mathbb{E}[Y \mid X] - f(X))^2\right] = 0 \\ \iff\ &\mathbb{P}(f(X) \neq \mathbb{E}[Y \mid X]) = 0 \end{aligned}$$

The ✗ version puts the first condition in words and the second in math. The ✓ version says the result once in words and once in math.

Give every relationship stated in words a formal expression

When a sentence states a mathematical relationship, also write that relationship as a formal expression. A relationship is any claim that links quantities, such as:

- an equality or an inequality;
- a limit;
- a proportionality;
- a bound or an ordering;
- a "depends on".

Keep the sentence, because it tells the reader what the relationship means, and add the displayed formula, because it tells the reader exactly what the relationship is. Prose alone leaves the reader to guess the quantities, the direction, and the conditions. Put every condition of the prose in the formula too, and introduce its symbols in the sentence. Keep the sentence and the formula as two sentences, as the rule to write each sentence in words or in math requires.

Example 0.5 (Adding the formula to a prose relationship)**.**

✗ If random variables X and Y are independent, the variance of their sum is the sum of their variances.

✓ If random variables X and Y are independent, the variance of their sum is the sum of their variances. In symbols:

$$X \perp\!\!\!\perp Y \implies \operatorname{Var}(X+Y) = \operatorname{Var}(X) + \operatorname{Var}(Y).$$

The ✗ version leaves the reader to reconstruct the formula and its condition. The ✓ version keeps the sentence and adds the relationship in math, with the independence condition inside the formula.

Introduce a theorem with a sentence

The line before a theorem, lemma or algorithm should be a complete sentence, or end with a colon. Do not let a heading such as "Theorem" finish a sentence that began above it.

Example 0.1 (Introducing a theorem)**.**

✗ This gives the following **Theorem.** $G(x)$ is increasing.

✓ Next we prove a result about monotonicity. **Theorem.** The function $G(x)$ defined in equation (3) is increasing.

The ✗ version breaks a sentence across the theorem and starts the theorem with a symbol. The ✓ version uses a complete sentence, and the theorem names its object.

Write the statement of a theorem so that it stands alone. A reader who jumps to the theorem should not need the assumptions in the paragraph before it, so repeat them in the statement.

Display and number the formulas that matter

Put each important formula on a line by itself. Number the important ones, even if you do not refer to them, because a reader may want to refer to them later. Quarto's equation cross-references number a display and let you link to it.

A key equation always goes in a display, never inline. A key equation is any of these:

- the conclusion of a theorem, lemma, or proposition;
- an equation that a proof builds toward;
- an equation that a later step refers to.

In a proof, a sentence such as "so $A = B$" or "this gives $A = B$" hides the result in the middle of a line. Put the equation in a display and let the sentence introduce it. Use inline math for symbols and short expressions inside a sentence, not for the equation the reader is meant to remember.

Example 0.1 (Displaying the normal equations)**.**

✗ Setting the gradient to zero yields $D^\top D\vec{\beta} = D^\top \underset{\sim}{y}$.

✓ Setting the gradient to zero yields the normal equations:

$$D^\top D\vec{\beta} = D^\top \underset{\sim}{y}$$

In the ✗ version the result sits in the middle of a line, where the reader can miss it. The ✓ version gives it a line of its own.

Write numbers according to how they are used

Spell out a small number when it is an adjective: "the proof uses three lemmas". Write the numeral when the number is a name or a value: "Lemma 3 is used 12 times", "the total rose by 4". Capitalize the name of a numbered result, as in "Theorem 1" and "Lemma 2".

Keep subscripts under control

Do not name the elements of a set unless you need to. If you write "Let $X = \{x_1, \dots, x_n\}$", you will need $x_{i_1}, \dots, x_{i_m}$ as soon as you take a subset of X. A subscript on a subscript is hard to read. Write "an element x of X" instead, and refer to x and y for two elements. Use x_1 and x_2 only when you need two specified elements.

Choose the voice and the person

Use “we” to mean the author and the reader together, as in “we can now prove”. Write “we prove”, not the passive “it can be proved”. Avoid “I” unless the author’s persona is relevant.

Write “assume that G is finite”, not “assume G is finite”. The word “that” helps the reader parse the sentence. Do not write “we have that $y = z$”; write “we have $y = z$”.

Choose the proof that fits the claim

Do not use proof by contradiction where a direct proof will do. The course’s worked example proves a lemma both ways. The direct proof is shorter, and it does not assume that the entries are integers or restate their ordering.

Cite results, not solutions

A proof, or any derivation that supports a result, never cites a solution to an exercise. A solution answers an exercise for the reader who attempted the exercise, so the solution is not a result that other arguments can build on. Follow each solution that contains a reusable result with a result div (a theorem, lemma, corollary or proposition) that states the claim. The test is whether the solution contains a result that other arguments could use, not whether any later argument currently relies on it. Cite that result div in every argument that uses the claim.

> **Example 0.1** (Citing a result div)**.**
>
> ✗ **Proof.** By Solution 9 and Solution 10, the claim holds.
>
> ✓ **Proof.** By Lemma 4 and Lemma 5, the claim holds.
>
> The ✗ version sends the reader to answers that the notes may reorder, hide or drop. The ✓ version cites claims that the notes state, number and keep.

Add an exercise and its solution just before every result div that you add or edit. The exercise does not state the result. It asks the student to find the result, such as “find an expression for the spread of a random variable using its first two moments”, so that the reader discovers the result, and the result div then states what the solution showed. Every later argument cites the result div.

The one exception to the citation rule follows from that pairing: the proof of the result itself may cite the solution of the exercise that asks for that same result. Such a proof does no derivation of its own: it is the one line `By @sol-x.`, with one citation. This is the one place

where a proof cites a solution. Derivation format describes the exercise, solution, theorem and proof pattern. When the exercise has several parts, the line may cite each part, as in the line `By @sol-x and @sol-x-part2.`, which ends in a period. A proof with its own steps or display math cites result divs instead.

The `check-solution-citations` workflow in `Morrison-Lab/gha` flags a `@sol-` cross-reference inside a proof or result div. It lets a one-line proof with one citation pass. For a one-line proof that cites several parts of the paired solution, the HTML comment `<!-- check-solution-citations: allow -->` on that line tells the workflow to skip that line. Quarto removes the comment from the rendered page. The workflow does not check prose outside proof and result divs, but the rule still applies there.

Keep each div minimal: cite, never inline

Every result div states one result and every solution div answers one exercise, and each does no more. When a proof or a solution needs a step that a reader could use elsewhere, state that step as its own result div, preceded by its exercise and solution, and cite that result div. Never derive the step inline. A solution decomposes the same way: each solution is the smallest argument that answers its exercise, and it cites result divs for the steps that other arguments also use.

Example 0.1 (Citing a step instead of deriving it)**.**

> ✗ **Proof.** By the chain rule, the gradient of each summand is $2\hat{\varepsilon}_i \underset{\sim}{D}(\underset{\sim}{x}_i)$, where $\hat{\varepsilon}_i$ is the residual and $\underset{\sim}{D}(\underset{\sim}{x}_i)$ the design vector, and summing over i gives the claim.
>
> ✓ **Proof.** Lemma 4 gives the gradient of each summand, and Lemma 5 says that the gradient of a sum is the sum of the gradients, so the gradient of the total is the sum of $2\hat{\varepsilon}_i \underset{\sim}{D}(\underset{\sim}{x}_i)$ over i.

The ✗ proof derives inline the result that Lemma 4 states, so a reader who needs that step elsewhere has nowhere to cite.

No div repeats reasoning that it could cite instead. Two arguments that share a step cite one result div, so a correction to the step is made once.

No check can decide whether a step is reusable, so this rule stays a writing rule that review enforces. The `check-solution-citations` workflow covers only the citation of solutions.

Use no symbol that the statement does not need

Halmos advises "avoid the use of irrelevant symbols" and states the rule in its showy form as "use no letter only once" (Halmos 1970, sec. 15). A symbol that appears once, in the sentence that introduces it, adds a name for the reader to hold and nothing to use it for. Delete the symbol, or use it in the statement that needs it.

Example 0.1 (Deleting a symbol that is named and never used)**.**

✗ Every continuous real-valued function f on a compact space is bounded.

✓ Every continuous real-valued function on a compact space is bounded.

Nothing in the sentence refers to f again, so the symbol contributes nothing to the statement. Halmos gives this sentence as his example (Halmos 1970, sec. 15).

When you apply a displayed relation to a second object, name the relation or restate it, rather than citing its tag. Halmos calls "g also satisfies $(\,)$" *"logical and alphabetical nonsense" and advises "*$(\,)$ remains satisfied if f is replaced by g", or, better, giving the relation a name (Halmos 1970, sec. 15). A named relation also lets the reader find it without hunting for a tag.

Example 0.2 (Naming the relation instead of tagging it)**.**

✗ The sample mean satisfies Equation 3, which says that the estimator's expected value equals the parameter. The sample variance also satisfies Equation 3.

✓ The sample mean is an unbiased estimator of the population mean. The sample variance is an unbiased estimator of the population variance.

In the ✗ version, "also satisfies Equation 3" is a statement about a different parameter, and the equation says nothing about which parameter. The ✓ version names the property, "unbiased", and states each parameter.

Halmos also says this rule "is sometimes better to break than to obey": a symbol that you introduce so that a later sentence in the same paragraph can use it is not superfluous. No check can tell a symbol that a later sentence uses from one that it never uses, because "later" can span pages, so this stays a writing rule that review enforces.

Repeat the wording of parallel results, then point out the difference

When a later result treats a parallel or more general case, Halmos advises that "it is a big help to the reader if you use the same words in the same order for as long as possible, and then, with a proper roll of drums, emphasize the difference" (Halmos 1970, sec. 11). He adds that listing the shared conditions and dropping one is not enough; say which condition changed (Halmos 1970, sec. 11).

Example 0.1 (Defining the lasso in the words of the ridge definition)**.**

> ✗ The ridge estimate minimizes the residual sum of squares plus a penalty on the size of the coefficients, which is the sum of their squares. The lasso estimate is the vector of coefficients with the smallest penalized fit, where the penalty adds up the absolute values.
>
> ✓ The ridge estimate minimizes the residual sum of squares plus a penalty equal to the sum of the squared coefficients. The lasso estimate minimizes the residual sum of squares plus a penalty equal to the sum of the absolute values of the coefficients. The two definitions differ only in the penalty.

The ✗ pair varies every phrase, so the reader must compare them clause by clause to find that only the penalty differs. The ✓ pair repeats the shared words and ends by naming the one change.

Rewording is for the places where the meaning differs. Do not repeat a result for the sake of repetition; reuse one written version of each result covers that case.

No check can decide that two results are parallel, so this stays a writing rule that review enforces.

State a theorem in one short sentence

Halmos advises that "the statement of the theorem should consist of one sentence whenever possible", and that the sentence be a short one (Halmos 1970, sec. 10). Put the introductory comments, preliminary definitions and motivation before the statement, and the proof after it. Leave asides out of the statement itself. Halmos says that phrases such as "without loss of generality we may assume …" and "moreover it follows from Theorem 1 that …" do not belong in the statement of a theorem (Halmos 1970, sec. 10). A reasoning word in a statement ("so", "because") is a separate fault, and Build every unit from smaller atomic units covers it.

Example 0.1 (Moving an aside out of a statement)**.**

✗ Without loss of generality, let $x > 0$. Moreover, the sequence converges.

✓ If $x > 0$, then the sequence converges.

The ✗ statement is two sentences, and the first one is an aside. The ✓ statement is one implication. If the case $x < 0$ needs the same argument, say so in the proof.

A statement that lists many hypotheses and many conclusions is too long for one sentence. Halmos says that "a list of eight hypotheses … and a list of six conclusions do not a theorem make; they are a badly expounded theory", and that the hypotheses "probably describe a general concept that deserves to be isolated, named, and studied" (Halmos 1970, sec. 10). Name the shared concept in its own definition, and state each conclusion as its own result.

A warn-only check (gha#1368) flags "without loss of generality", "moreover" and "furthermore" on a line that a pull request adds to a theorem statement. No check can measure whether a statement is short or whether its hypotheses form a concept, so the length rule stays a writing rule that review enforces.

Rules that other chapters cover

Table 1 lists the course's other rules and where this book discusses them.

Table 1: Rules from the course that other chapters of this book cover

Rule	Where it appears
Define every variable when you introduce it	Defining terms
Use one notation for one thing, and one thing for each notation	Mathematical notation
Motivate a result before you derive it	Show why you need a result before you derive it
Avoid strings of nouns used as adjectives	Rewrite noun strings and hidden verbs
Avoid unnecessary jargon	Minimize unnecessary jargon

Rule	Where it appears
Write sentences that can be read from left to right without ambiguity	Avoid garden-path sentences
Use "that" instead of "which" in a restrictive clause	Relative pronouns
Do not praise your own work	Tone

Halmos, Paul R. 1970. "How to Write Mathematics." *L'Enseignement Mathématique* 16. https://doi.org/10.5169/seals-43857.

Knuth, Donald E., Tracy Larrabee, and Paul M. Roberts. 1987. *Mathematical Writing*. https://jmlr.csail.mit.edu/reviewing-papers/knuth_mathematical_writing.pdf.