MediTrace — Statistical Methodology

Pharmacology-based effect analysis for medications and supplements

SilvaGIS GmbH · Dumeni Cavegn · vitatrace.app · as of June 2026 · app version 1.1.7

> **Important note:** MediTrace provides statistical correlation indicators — not a medical diagnosis and not a treatment recommendation. Correlation is not causation. Any changes to medication must be discussed with a doctor. The app is **not a regulated medical device**.

1. Overview: two analysis methods + two side analyses

MediTrace relies on **one clear, statistical analysis**. The previously offered "quick analysis" (a simple mean comparison) was removed — it was weaker and could contradict the scientific analysis. Only the robust evaluation remains. All methods operate on **custom fields**: freely definable symptom or measurement scales (e.g. energy 0–10, pain 0–10).

Method	Use case	Principle	Min. data
Ordinal logistic regression (OLR)	As-needed and short-acting medications	Proportional-odds model with pharmacologically driven lag search; optional intra-day resolution	≥ 5 days with + 5 without intake
Long-term trend	Daily medications with a build-up phase (buildupDays > 7), e.g. sertraline, vitamin D	Baseline / build-up / steady state — phase comparison	≥ 7 entries

Independently, two **side analyses** run when the required data is available:

- **Vital-sign effect analysis** — effect of a medication on vital measurements (blood pressure, pulse), see section 6
- **Dose-response analysis** — with several documented doses, see section 7

2. Pharmacological lag database

MediTrace's central differentiator: every medication acts within a different time window. A simple correlation of "taken today → symptom today" would be scientifically worthless for antidepressants or vitamins — their effect only appears weeks later. MediTrace therefore stores pharmacological profiles for **448 medications and supplements**.

Parameter	Meaning	Ibuprofen	Cetirizine	Sertraline	Vitamin D3
`onsetHours`	Onset of action	0.5 h	1 h	4 h	24 h
`peakHours`	Peak effect	2 h	4 h	8 h	168 h
`durationHours`	Duration of effect	6 h	24 h	24 h	336 h
`buildupDays`	Build-up time	—	—	28 days	60 days

For medications without a stored profile, fallback values per category are used (analgesics, antihistamines, antidepressants, vitamins, etc.).

Lag-window calculation

> Tested lag windows for short-acting medications:
> $$\text{lags} = \{\text{onsetHours}, \text{peakHours}, \text{durationHours}, 2\times\text{durationHours}, 48\text{h}, 72\text{h}\} \cap [0, 2\times\text{durationHours}]$$

> Fine lag windows (intra-day) when duration < 24 h and hourly resolution is viable:
> $$\text{lags} = \{1\text{h}, 2\text{h}, 3\text{h}, \ldots, 12\text{h}\}$$

> Tested lag windows for long-acting medications (buildupDays > 7):
> $$\text{lags} = \{0.25\times\text{buildupDays}, 0.50\times\text{buildupDays}, 0.75\times\text{buildupDays}, 1.0\times\text{buildupDays}\}\ [\text{in days}]$$

All lag windows are tested; the lag chosen is the one with the highest **signal strength** (largest field effect), not by model accuracy alone — so the pharmacologically most plausible time lag is preferred.

3. Data preparation and normalization

For each custom field, every daily value is normalized to [0, 1], taking the scale direction into account:

> Normalized main value per entry (averaged over all custom fields):
> $$\text{norm}(v, \text{field}) = (v - \min) \,/\, (\max - \min) \text{ if field.direction} = \text{'higher\_is\_better'}$$
> $$\text{norm}(v, \text{field}) = 1 - (v - \min) \,/\, (\max - \min) \text{ if field.direction} = \text{'lower\_is\_better'}$$
>
> $$\text{mainValue}(\text{entry}) = \sum \text{norm}(v_i) \,/\, \text{n\_fields}$$

Example: pain scale 0–10 with `lower_is_better`: a value of 3 yields norm = 1 − 3/10 = 0.7 (high value = good, because little pain).

For the OLR, the normalized mainValue is compressed into 5 ordinal classes:

> Class assignment ($K \in \{0, 1, 2, 3, 4\}$):
> $$K = \text{floor}(\text{mainValue} \times 5), \text{ where } K = 4 \text{ if mainValue} = 1.0$$

4. Ordinal logistic regression (main analysis)

For as-needed and short-acting medications, MediTrace uses a proportional-odds model (cumulative-link model) that models the relationship between intake and well-being on the ordinal 5-class variable — while simultaneously accounting for confounders.

4.1 Model equation

> Proportional-odds model (for each threshold k = 0..3):
>
> $$P(Y \le k \mid x) = \sigma(\theta_k - \beta \cdot x)$$
>
> $$\sigma(z) = 1 / (1 + e^{-z}) \text{ [sigmoid function]}$$
>
> $$\beta \cdot x = \beta_1 \cdot \text{intake} + \beta_2 \cdot \text{dose} + \beta_3 \cdot \text{weekday} + \beta_4 \cdot \text{otherMeds}$$

4.2 Input variables (4 features)

- **intake**: 0/1 — substance taken on this day (with lag L)
- **dose**: normalized dose relative to the standard dose (e.g. 1.5× = 1.5) — enables the dose-response effect
- **weekday**: weekday / 6 (0 = Monday … 1 = Sunday) — confounder for the weekly rhythm
- **otherMeds**: number of other active medications on the same day / 10 (max. 1) — confounder for parallel intakes

At intra-day resolution (hourly resolution, see 4.4) the dose differentiation is dropped; the remaining features stay.

4.3 Loss function and optimization

> Negative log-likelihood (cumulative-link):
>
> $$L(\theta, \beta) = -\sum_n \sum_k 1[y_n = k] \cdot \log[P(Y = k \mid x_n)]$$
>
> $$P(Y = k \mid x) = \sigma(\theta_k - \beta \cdot x) - \sigma(\theta_{k-1} - \beta \cdot x)$$

Minimized via gradient descent (400 epochs, learning rate 0.3) with L2 regularization (λ = 0.01) and gradient clipping (±5). Threshold monotonicity is enforced after each step: $\theta_0 \le \theta_1 \le \theta_2 \le \theta_3$. Because the model is fitted iteratively, there is — unlike a t-test — no closed-form hand formula for the coefficients; what is reported are the resulting field effects and a confidence value.

4.4 Intra-day resolution

When enough days have several timestamped ratings (condition `hourResolutionViable`: ≥ 30% of days multi-rated and timestamps present) and the duration is < 24 h, the app compares *within* a day — e.g. "before effect" vs. "at peak". This reveals an effect even for a medication taken daily.

4.5 Confidence calculation

> Weighted confidence score (0–100%):
>
> $$\text{signalStrength} = \min(\max(|\text{field effect}|) / 5, 1)$$
> $$\text{dataFactor} = \min(\text{dataPoints} / 30, 1)$$
> $$\text{olrFactor} = \min(\text{bestLag.accuracy} \times 2, 1)$$
>
> $$\text{confidence} = (\text{signalStrength} \times 0.6 + \text{dataFactor} \times 0.25 + \text{olrFactor} \times 0.15) \times 100$$

Confidence	Level	Meaning
< 40%	`not_evaluable`	Too little data or no pattern detectable
40–60%	`preliminary`	First indication — more tracking days recommended
> 60%	`reliable`	Robust statement

5. Long-term trend: phase analysis

For medications with a long build-up phase (buildupDays > 7), a daily comparison is unsuitable — the steady-state effect only sets in after weeks. MediTrace therefore splits the data into three phases:

> **Baseline:** all entries *before* the intake start date

> **Build-up phase:** entries from the start date to start date + buildupDays

> **Steady state:** entries after start date + buildupDays

> Before/after difference:
>
> $$\text{diff} = (\mu\_\text{steadyState} - \mu\_\text{baseline}) \times 4$$
> $$\text{effectSize} = |\text{diff}| \,/\, 2$$
> $$\text{dataFactor} = \min(\text{dataPoints} \,/\, 30, 1)$$
> $$\text{accuracy} = \min(0.5 + \text{effectSize} \times 0.4, 0.95) \times \text{dataFactor}$$
> $$\text{confidence} = \text{round}(\text{accuracy} \times 100 \times (0.6 \text{ if still in build-up phase, otherwise } 1.0))$$

Fallback when no baseline exists (tracking started only after intake began): first 7 entries as baseline, last 14 entries as the comparison group.

The PDF report shows a **build-up curve** with the onset-of-effect time point computed from the data (intersection of the trend line with the baseline + 0.3 threshold).

6. Vital-sign effect analysis

In addition to the symptom analysis, MediTrace checks how a substance affects recorded **vital measurements** — so unwanted effects become visible too, not just the intended one. It compares the measurements on days *with* vs. *without* intake.

> Effect on a measurement:
>
> $$\text{diff} = \varnothing\text{measurement}(\text{with intake}) - \varnothing\text{measurement}(\text{without intake})$$
> $$\text{Minimum data: } \geq 3 \text{ measurements per group}$$

An effect is only reported if it exceeds a clinically relevant minimum threshold:

Measurement	Minimum difference for output
Blood pressure, systolic	5 mmHg
Blood pressure, diastolic	4 mmHg
Pulse	4 bpm

Example output: *"Ibuprofen raises systolic blood pressure by ~11 mmHg on average"* or *"Amitriptyline raises resting pulse by ~12 bpm on average"*. The analysis only describes existing data — it is not a

recommendation on dosage or discontinuation.

7. Dose-response analysis

If several different doses were documented (≥ 2 distinguishable dose levels), MediTrace additionally computes a dose-response trend:

> Normalized dose intensity:
> $\text{intensity}(d) = (d - d\_min) \,/\, (d\_max - d\_min)$ [$0 = \text{lowest}, 1 = \text{highest documented dose}$]

The app shows the mean effect values per dose level and a trend (increasing / flat / mixed). With too few data points per level, "not yet evaluable" is reported.

8. Worked example: ibuprofen for headache (OLR)

Scenario: 30-day diary, custom field "headache" (0–10, `lower_is_better`), custom field "energy" (0–10, `higher_is_better`). Ibuprofen as needed, partly 400 mg, partly 600 mg. Ibuprofen profile: onset 0.5 h, peak 2 h, duration 6 h → short-acting → **OLR**.

8.1 Simulated data (excerpt)

Date	Ibuprofen	Dose	Headache	Energy	mainValue
Day 1	✓	400 mg	7	4	0.35
Day 2	—	—	3	7	0.70
Day 3	✓	600 mg	8	3	0.25
Day 4	—	—	2	8	0.80
Day 5	✓	400 mg	6	5	0.45
… 25 more days (12 with, 13 without) …					
ø WITH			6.8	4.2	**0.33**
ø WITHOUT			2.5	7.5	**0.75**

8.2 Model and field effects

The app builds a feature vector [intake, dose, weekday, otherMeds] per day and fits the OLR model iteratively (see section 4). What it reports are the **field effects** — the difference in field values between intake and non-intake days, at the best lag:

Field	ø WITH	ø WITHOUT	Difference	Scale direction	Interpretation
Headache	6.8	2.5	+4.3	lower_is_better	Pain higher on intake days → medication taken during strong pain
Energy	4.2	7.5	−3.3	higher_is_better	Energy lower on intake days → headache days are low-energy anyway

8.3 Confidence

> $\text{signalStrength} = \min(4.3 \,/\, 5, 1) = 0.86$
> $\text{dataFactor} = \min(25 \,/\, 30, 1) = 0.83$
> $\text{olrFactor} = \min(0.45 \times 2, 1) = 0.90 \;\; (\text{example model accuracy})$
> $\text{confidence} = (0.86 \times 0.6 + 0.83 \times 0.25 + 0.90 \times 0.15) \times 100 \approx 86\% \rightarrow \text{"reliable"}$

Important confounding note: for as-needed pain medications the correlation is expectedly "negative" — intake happens *because* there is pain, not the other way around. The app states this explicitly: "intake and

pain occur together (as expected for an as-needed medication)" — not "the medication causes pain". The honest assessment of benefit belongs in the consultation.

9. Worked example: sertraline (long-term antidepressant)

Sertraline profile: buildupDays 28, onset 4 h, duration 24 h → daily with build-up phase → **long-term trend**. 60 days of tracking, intake start on day 15.

> **Baseline (days 1–14):** n = 14, μ_mainValue = 0.38 (low energy, high exhaustion)

> **Build-up phase (days 15–42):** n = 28, μ = 0.44 (slight improvement)

> **Steady state (days 43–60):** n = 18, μ = 0.61 (clear improvement)

$$\text{diff} = (0.61 - 0.38) \times 4 = +0.92$$
$$\text{effectSize} = 0.92 \,/\, 2 = 0.46$$
$$\text{dataFactor} = \min((14 + 18) \,/\, 30, 1) = 1.0$$
$$\text{accuracy} = \min(0.5 + 0.46 \times 0.4, 0.95) \times 1.0 = \mathbf{0.684}$$
$$\text{confidence} = \text{round}(0.684 \times 100 \times 1.0) = 68\% \rightarrow \text{"reliable"}$$

From the 28 build-up-phase entries, the app computes a linear trend; the intersection with the threshold (μ_baseline + 0.3) yields the computed onset of effect — here approx. day 22 after intake start (within the 2–4-week rule of thumb for SSRIs).

> **Important:** the computed onset of effect is a statistical indicator based on self-assessment — not a pharmacological diagnosis. It serves as a basis for discussion with the treating professional.

10. Significance criteria and limitations

An effect is reported as "significant / reliable" when **both** conditions are met:

- **Effect size:** a noticeable field difference (guideline |diff| ≳ 1.5 points on the 0–10 scale)
- **Confidence:** ≥ 60% (level "reliable")

Known limitations

- **Confounding:** if several medications are regularly taken at the same time, statistics cannot cleanly distinguish which one is working. The `otherMeds` feature mitigates this but does not fully solve it. The app flags such cases.
- **Reverse causality for as-needed medication:** intake happens because of the symptom — the correlation direction is then expectedly negative and is communicated as such.
- **Minimum data:** OLR ≥ 5 days with + 5 without intake; long-term trend ≥ 7 entries plus a steady-state phase; vital/dose analyses ≥ 3 or ≥ 2 levels.

- **Pharmacokinetic database:** active-ingredient profiles are reference values from published sources — not official prescribing information; individual variation is possible.
- **Self-report:** data are self-recorded (recall and confirmation bias possible).

11. Privacy and architecture

- **Offline-first:** all data (intakes, well-being, measurements, analysis results) reside exclusively in a local SQLite database on the device.
- **No telemetry:** no analytics SDKs, no crash reporter with PII, no cloud sync.
- **Network calls:** only two — RevenueCat (encrypted entitlement status) and an app-store update check. No health data leaves the device.
- **PIN + biometrics:** app lock via Secure Store, optional but enabled by default.
- **Backup:** encrypted JSON export for your own safekeeping — no recovery without this file.

12. Source-code references

The implementation lives in the following TypeScript files:

- `src/data/medications-db.ts` — pharmacological database (448 entries) + lag calculation
- `src/services/ordinalLogit.ts` — ordinal logistic regression + lag optimization + intra-day
- `src/services/longtermAnalysis.ts` — long-term trend + phase split
- `src/services/vitalAnalysis.ts` — vital-sign effect analysis (blood pressure, pulse)
- `src/utils/analysisUtils.ts` — normalization, field effects, dose-response calculation
- `src/services/analysisEngine.ts` — confidence levels, lag constants

I provide the source code on scientific request.
Contact: **vitatrace@proton.me**
