METHODOLOGY WHITE PAPER · v1.0

How IR Buddy produces defensible incidence estimates.

IR Buddy converts a researcher's plain-English audience description into a structured brief, dispatches criterion-scoped research agents against a category-routed registry of Tier 1–4 U.S. sources, and returns a single incidence-rate estimate assembled by deterministic math — not by a language model guessing a number. This paper documents the five stages of that pipeline: intake & parse, criterion classification and source routing, per-criterion retrieval with a judge/retry loop, deterministic assembly, and static deliverable production.

The deterministic boundary

IR Buddy draws a hard line between where language models operate and where they do not. LLMs are permitted only to (a) parse free text into a controlled taxonomy, (b) classify a criterion into its research category so the correct approved-source allowlist is chosen, (c) extract prevalence values from retrieved source documents, and (d) generate the human-readable prose around the numbers in the final report. Every prevalence figure, every joint probability, every adjustment factor, and every final incidence percent is produced by lookup against a cited source or by explicit arithmetic in a Code node. If a figure cannot be sourced at the requested granularity, the pipeline marks the criterion UNVERIFIABLE — it does not synthesize a number.

Pipeline at a glance

1. Intake & Parse	2. Classify & Route	3. Retrieve & Judge	4. Assemble	5. Deliver
Website wizard → n8n webhook. Slim prompt-creator Code node emits text, targets, preservedContext.	LLM classifier tags each criterion (Demographic, Health, Firmographic, Behavior, Attitudinal). Approved-source allowlist selected from registry.	Per criterion: Perplexity Deep Research with search_domain_filter OR Tavily+GPT-4o-mini. JSON-schema output. Judge-loop retries with Tier 2 fallback.	Code node: IR = joint × marginals × recency × correlation × panel_deflation. Confidence + planning interval assigned.	Deterministic PDF builder renders report_data → branded PDF email to researcher. Lead + result logged to Airtable.

Stage 1 — Intake & slim parse

The homepage wizard collects six items: work email, sample type, age bands, gender, geography, and a taxonomy palette spanning twelve dimensions (age, gender, ethnicity, HHI, education, employment, parental status, age of child, marital status, home ownership, auto ownership, and common disease state), plus a freeTextCriteria box for anything the palette doesn't cover. Every value carries a stable valueCode; the payload is the ground truth every downstream node reads.

The n8n Prompt Creator Code node then produces a slim contract — not a mini-brief. Its output is exactly three keys: a short text field (one clean sentence: “Among the targetable population of [targets], research the prevalence of [freeTextCriteria].”), a targets array of the selected chip labels, and the raw preservedContext. Structured context lives in the HTTP body, not inside the prompt string — that separation is the single biggest reliability lever in the pipeline, because it eliminates the double-encoded JSON that used to fool Perplexity into searching for the word “target” and returning Target Corporation results.

LLM role in Stage 1	Parse only — free text → taxonomy valueCodes. Temperature 0, JSON-schema constrained. Never asked for a number.
Prompt contract	Slim text + targets + preservedContext. Query intent, structured context, and output-format rules kept in three separate places.
Failure mode	Novel criterion flagged; parser never invents a valueCode. Downstream nodes handle the flag explicitly.

n8n node topology

The Stage 1 payload flows through a fixed n8n graph: Webhook → Parse Criteria LLM (classify + normalize free text) → Clean/Parse Code node → Split In Batches (one item per criterion) → IRB Prompt Creator Code node (text, targets, preservedContext) → Build Perplexity Body Code node (assembles the entire HTTP body as JSON literal so the request node uses a single {{ $json }} expression) → HTTP Request to Perplexity Deep Research (or the Tavily+4o-mini branch) → Result Parser Code node (strips fences, extracts JSON, validates schema) → Merge by index to recover the original criterion and preservedContext → IF (>=2 usable estimates?) → Attach Match to Criterion → Aggregate → Deterministic IR Code node → PDF Builder Code node → Send Email → Airtable Log. Every node is idempotent on the criterion index, which is what makes the retry branch safe.

Stage 2 — Classify the criterion, route to approved sources

Stage 4 of the old design — "send everything to Sonar" — failed on non-health criteria. Deep Research runs a multi-pass agentic search per call; for well-known health statistics (e.g., preterm birth rates from CDC) that is fine, but for "avid sports fans," "Toyota non-rejectors," or "$3,500+/month card spenders" there is no single authoritative federal source. The fix is not tighter prompting — it is source-routed retrieval. An LLM classifier tags each criterion; the classifier's only output is the criterion category, which selects a hardline domain allowlist from an Airtable-backed source registry before any search runs.

Category	Subcategory	Tier 1 allowlist
Demographic	Age, gender, race, ethnicity	census.gov, data.census.gov, bls.gov
Demographic	Income, employment, occupation, industry	census.gov, bls.gov, bea.gov
Demographic	Household, marital, parental, housing tenure	census.gov (ACS, CPS), sipp data
Health	Maternal / infant / birth outcomes	cdc.gov, nchs.cdc.gov, nih.gov, pubmed.ncbi.nlm.nih.gov, marchofdimes.org
Health	Chronic conditions, prevalence, diagnosis	cdc.gov, nih.gov, nhlbi.nih.gov, nci.nih.gov, pubmed.ncbi.nlm.nih.gov, ahrq.gov, cms.gov
Firmographic	Employer size, industry, establishments	census.gov, bls.gov, sba.gov
Attitudinal	Political, tech adoption, opinion, values	pewresearch.org, apnorc.org, gallup.com, ssrs.com
Behavior	Media, purchase, category penetration	Tier 2 syndicated: MRI-Simmons, GfK MRI, Scarborough

The registry supports up to 20 domains per request (Perplexity's search_domain_filter cap). The classifier can also emit a hybrid route (e.g., a health-adjacent behavior criterion) that carries two allowlists in sequence.

Stage 3 — Per-criterion retrieval, judge, and retry

Criteria fan out through a Split In Batches node so each runs independently. By default every criterion calls Perplexity Deep Research with the category allowlist; the judge loop escalates to a Tavily+GPT-4o-mini branch when Deep Research fails to return a source-grounded number or when raw PDF-table extraction is required.

Path	When to use	Cost / latency	Quality profile
Perplexity Deep Research (default)	Every criterion by default. Multi-pass agentic retrieval across the category allowlist, with reasoning over intermediate findings.	~$0.20–0.60 · 30–90s	Highest recall on obscure or non-obvious figures; native support for search_domain_filter and structured JSON output.
Tavily Adv. + GPT-4o-mini (fallback)	Judge-loop escalation when Deep Research returns no source-grounded number, or when raw PDF-table extraction is needed.	~$0.017–0.03 · 8–15s	Tavily fetches raw content from the approved allowlist; 4o-mini extracts numeric prevalence into the schema.

Judge-loop & retry policy

The response goes to a lightweight parse/judge Code node immediately after the LLM. It (1) strips code fences, (2) parses the first JSON object out of choices.message.content, (3) validates against the per-criterion schema, and (4) checks a pass/fail rubric: prevalence is a decimal in [0, 1], at least one source URL resolves to the approved allowlist, and at least two independent estimates were returned when possible. On failure, an IF node routes back to the retriever once with a broadened Tier 2 allowlist appended. A second failure marks the criterion UNVERIFIABLE and continues — no infinite loops.

Per-agent JSON schema

Every retrieval agent returns the same shape so the assembler is a pure function of its inputs. The agent field varies with route; everything else is fixed.

```
{ agentType, criterionResearched, estimates: [ { sourceName, articleTitle, sourceUrl, prevalence, year, tier, population, assumption, notes } ], recommendedMid, confidence: HIGH | MEDIUM | LOW | UNVERIFIABLE, notes }
```

Token & quality controls

- Scoping. Each classifier route loads only the sources and lookup patterns for its category — ~60–80% context reduction versus a monolithic prompt.
- Slim prompt contract. Text = one strict sentence. All demographics, taxonomy, and metadata live in the HTTP body, never inside the prompt string.
- Tier-first stopping. Agents stop at the shallowest tier that satisfies granularity; only a validation failure triggers the Tier 2 fallback pass.
- Structured output only. The parse node discards any free-form prose the model emits; only schema-valid JSON survives into Stage 4.
- Sourced-answer contract. Every prevalence value must arrive with a source URL, publication year, tier, and population description. A response without a citable source is a failure, not a soft warning.

Source scoring rubric

Every returned estimate carries a numeric source score (0–100) that surfaces in the final report so the researcher can defend the choice. Score components: Tier (1 = 40 points, 2 = 25, 3 = 15, 4 = 5) · Recency (≤5y = 20, ≤10y = 10, older = 0) · Geography match (U.S. national at required granularity = 20) · Population match (exact target = 10, adjacent = 5) · Independence bonus (distinct sponsor from other cited estimates = 10). A criterion with an averaged score below 40 downgrades the overall confidence tier one step regardless of agreement.

Stage 4 — Deterministic assembly

Assembly is a pure function of the agent outputs, executed in an n8n Code node. Each criterion arrives as a population share p_j in [0, 1] with a source and geography stamp. Where a joint distribution is available (e.g., age × gender × ethnicity from ACS PUMS), the joint value is used directly. Where only marginals exist, criteria are combined multiplicatively and adjusted by three optional factors, each drawn only from published data — never estimated by an LLM.

Simplified formula

$$IR = \left(\prod p_{joint}\right) \times \left(\prod p_j^{marginal}\right) \times \alpha_{recency} \times \alpha_{correlation} \times \alpha_{panel}$$

$\alpha_{recency}$ discounts figures older than a category-specific half-life (e.g., 5 years for demographics, 3 for chronic prevalence, 1 for behavior). $\alpha_{correlation}$ corrects for known co-occurrence between criteria when a joint is unavailable but a published correlation is (e.g., income × education). α_{panel} deflates for the well-documented gap between census prevalence and panel-reachable prevalence for the target sample type. Every adjustment applied is enumerated in the final report so the researcher can defend it to a client.

Worked example

Target: U.S. women 25–44 who have delivered a preterm infant in the last five years, household income $75K+. The classifier tags four criteria — female (demographic), age 25–44 (demographic), preterm delivery in last 5 years (health / maternal), HHI ≥$75K (demographic). Age × gender comes from ACS PUMS as a joint (0.161). HHI ≥$75K among that age × gender cell comes from CPS ASEC (0.462). Preterm delivery rate among births in last 5 years comes from CDC/NCHS (0.101). Because CDC reports birth rate and not had a birth in last 5 years, the assembler applies a second lookup for “share of women 25–44 with a live birth in last 5 years” (≈0.38, ACS fertility supplement). α_{panel} = 0.85 for maternal in consumer panels. IR = 0.161 × 0.462 × 0.38 × 0.101 × 0.85 ≈ 0.24%; planning interval 0.18%–0.32%; confidence HIGH.

Planning interval & confidence

IR Buddy does not fabricate a formal statistical confidence interval across heterogeneous sources. Instead, each report ships a planning interval (low / mid / high) computed from the range of accepted source estimates plus a category-specific dispersion factor, and a categorical confidence tier:

Tier	Meaning
HIGH	All or nearly all criteria supported by Tier 1–2 sources with strong agreement.
MEDIUM	Mix of strong and moderate sources, or one criterion on lower-tier but credible evidence.
LOW	Key criteria rely on weak, old, sparse, or conflicting evidence.
UNVERIFIABLE	At least one required criterion cannot be credibly sourced at the requested granularity.

Stage 5 — Static deliverable, Airtable log

The assembler emits a single report_data JSON object — the source of truth for the deliverable. A pure-code PDF builder renders that object into the branded IR Buddy report template (masthead, targeting sentence, single IR headline with the teal band between low and high, criteria breakdown, full source table with plain-English source name, article title, URL, extracted IR, assumption, tier, and source score, and the recommendation block). No LLM touches the numbers at this stage. A second, tightly-templated LLM pass generates only the surrounding prose — the executive summary sentence, the criterion narrative, and the feasibility recommendation — and is forbidden from restating, rounding, or reinterpreting any figure.

The PDF is emailed to the researcher's work inbox, typically within two minutes of submission. In parallel, the lead and result are written to Airtable (base IR Buddy – Runs) with the full payload, chosen route per criterion, source URLs, and the assigned confidence tier — giving Andrew a live record for feedback and iteration.

What IR Buddy does not do

- Estimate a number when no source exists at the requested granularity.
- Combine criteria for which no joint distribution or published correlation is available without flagging the assumption.
- Let a language model guess a prevalence, adjust a figure, or invent a source URL.
- Cover geographies outside the United States in v1.
- Fabricate a formal statistical confidence interval across heterogeneous sources.

For the live estimator, sample report, and current source register, visit irbuddy.io. Methodology questions: reply to any IR Buddy report email.