The Eigenvector That Saved Birmingham

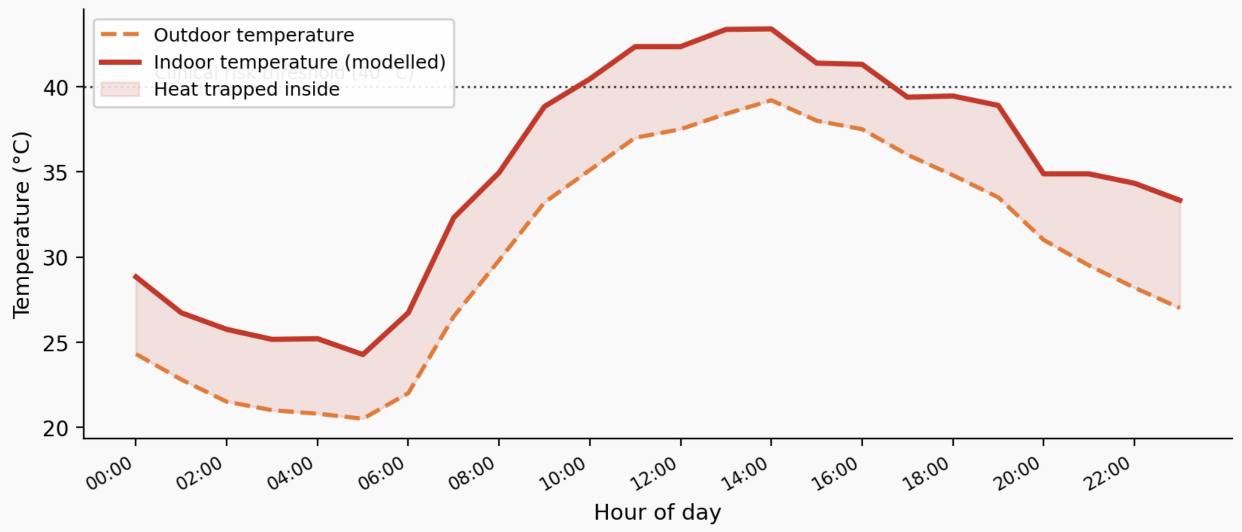
How a 45-year-old piece of linear algebra can decide

who gets the last ambulance during a heatwave

I spent a week-end modelling what happens inside Birmingham homes during a heatwave: solar gain through windows, thermal mass, insulation quality, humidity, wind speed, and building age. I ran the model against real data from 19 July 2022 — the day Birmingham hit 40 °C for the first time ever — and the output stopped me cold.

Indoor temperature peaked at 43.4 °C at 14:00. Three degrees above the outdoor maximum. The home, the supposedly safe place, was hotter than the street.[1]

Figure 1 — Modelled indoor vs outdoor temperature, Birmingham, 19 July 2022.
The home was 3 °C hotter than the street by mid-afternoon.



Look at the shaded gap in Figure 1. That is heat being actively trapped — the same insulation that works for you in January working against you in July. It created an immediate, ugly problem. Birmingham city authorities had finite ambulances, finite cooling centres, and finite emergency welfare teams. Across ten constituencies and roughly a million people. The question was: who goes first?

That sounds political. Maybe ethical. It is both. But it turns out it is also — and this is the part I did not see coming — a problem about eigenvectors.

The Problem with Trusting Your Gut

Four key factors drive heatwave vulnerability: the proportion of seriously disabled or ill residents (D), the proportion in vulnerable age groups — under 15 or over 65 — (A), average household income (I), and housing overcrowding (C). All four matter. The question is by how much each one matters relative to the others.

Try ranking all four at once. Is D three times more important than A, or five times? If D is six times more critical than I and A is five times more critical than I, then consistency demands D be only about 1.2 times more critical than A — yet your gut probably says closer to two. That quiet contradiction — invisible to you, but detectable by mathematics — is exactly the problem. In 1980, Thomas Saaty had a beautifully simple fix: instead of ranking everything at once, ask every possible *pair* of questions separately, and let the maths do the aggregating. He called it the Analytic Hierarchy Process — and Figure 2 shows how it structures the problem.

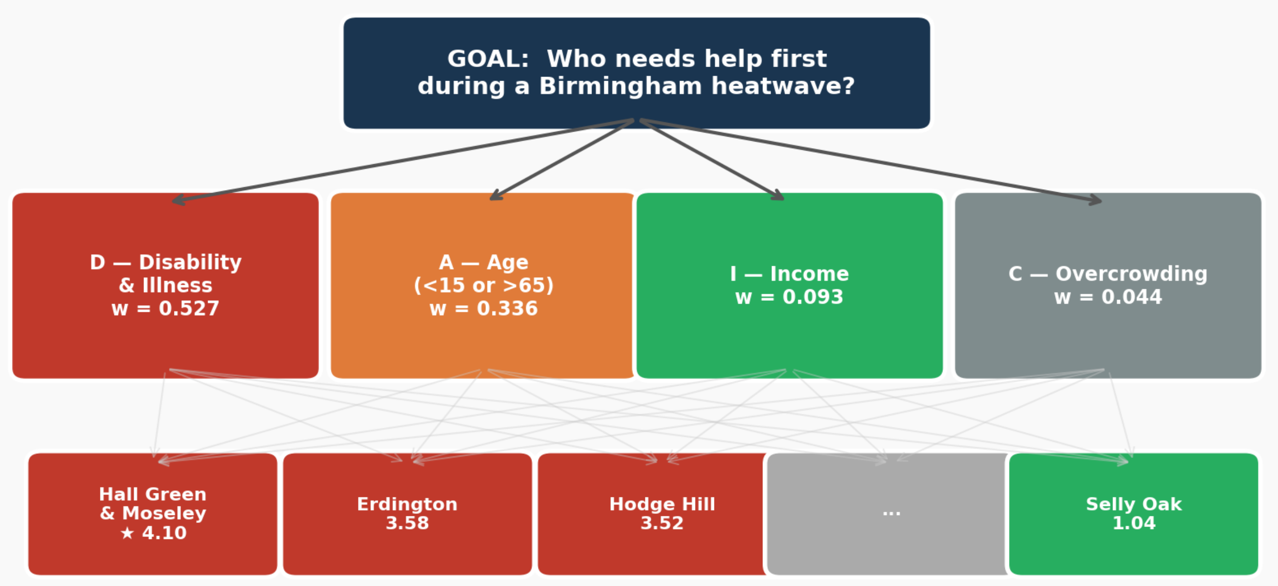


Figure 2 — AHP decision hierarchy: goal → criteria (with eigenvector weights) → ranked constituencies

Building a Matrix of Human Judgement

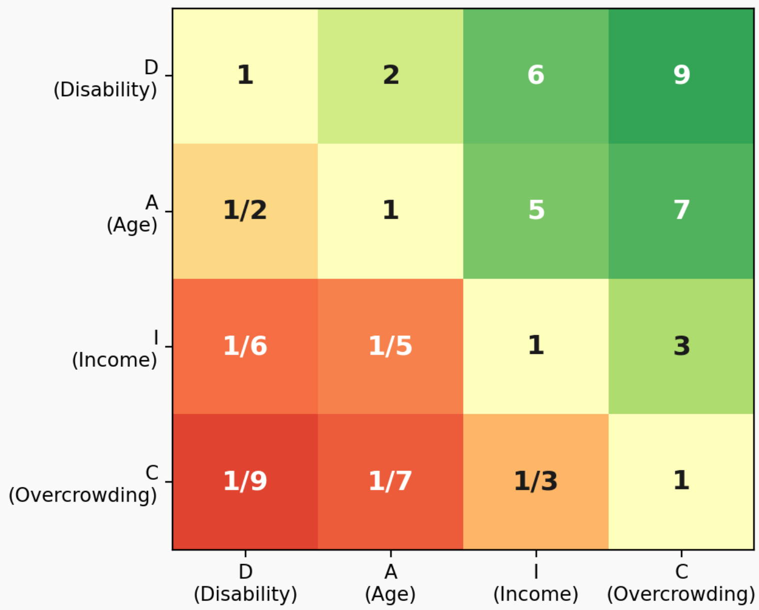
Saaty defined a nine-point scale for pairwise comparisons.[2] A score of 1 means equal importance; 3 is moderate; 5 strong; 7 very strong; 9 extreme. Fractions work in reverse: if D is six times more important than I, then I scores 1/6 against D. Reciprocals are built into the structure by design.

For Birmingham, I judged disability and illness (D) most critical: many affected residents rely on electrically powered medical equipment that fails during power cuts. Age came second — the young lose heat poorly, and the elderly are acutely vulnerable. Income came third. Overcrowding last. Encoding these pairwise judgements into a 4×4 matrix M, where entry M[i][j] records how much more important criterion i is than j:

$$
\begin{array}{c|cccc}
 & D & A & I & C \\
\hline
D & 1 & 2 & 6 & 9 \\
A & 1/2 & 1 & 5 & 7 \\
I & 1/6 & 1/5 & 1 & 3 \\
C & 1/9 & 1/7 & 1/3 & 1
\end{array}
$$

M is reciprocal: M[i][j] = 1/M[j][i]. The green top-right corner (Figure 3) shows D and A dominating; the dark-red bottom-left shows C being outweighed everywhere. Sixteen numbers — but still not a ranking. For that, you need an eigenvector.

Figure 3 — Pairwise comparison matrix M (green = row criterion more important; red = less)



The Eigenvector Hidden in the Matrix

Here is the part that genuinely got me. Suppose a perfect set of weights existed — a vector w = [w_D, w_A, w_I, w_C] capturing how important each criterion truly is. A perfectly consistent matrix would satisfy $Mw = \lambda \cdot w$ for some scalar λ. In other words: the weights we are looking for

are already hiding in the matrix as its eigenvector. Saaty proved that even when M is imperfectly consistent — as all real human judgements will be — the best possible weights are still given by the principal eigenvector of M: the one corresponding to the largest eigenvalue. And you can approximate it with nothing more than basic arithmetic.

✦ *Try this yourself — it takes five minutes and a calculator.*

Step 1: Normalise each column of M by dividing every entry by its column total.

Column D: $1 + 1/2 + 1/6 + 1/9 = 18/18 + 9/18 + 3/18 + 2/18 = 16/9 \approx 1.78$
Normalised column D: [0.562, 0.281, 0.094, 0.062]

Column A: $2 + 1 + 1/5 + 1/7 \approx 3.34$ → *normalised:* [0.598, 0.299, 0.060, 0.043]

Column I: $6 + 5 + 1 + 1/3 \approx 12.33$ → *normalised:* [0.486, 0.405, 0.081, 0.027]

Column C: $9 + 7 + 3 + 1 = 20$ → *normalised:* [0.450, 0.350, 0.150, 0.050]

Step 2: Average each row across all four normalised columns.

$w_D = (0.562 + 0.598 + 0.486 + 0.450) / 4 = 0.524$
$w_A = (0.281 + 0.299 + 0.405 + 0.350) / 4 = 0.334$
$w_I = (0.094 + 0.060 + 0.081 + 0.150) / 4 = 0.096$
$w_C = (0.062 + 0.043 + 0.027 + 0.050) / 4 = 0.046$

Repeat a few more times and the values converge. The true eigenvector is D=0.527, A=0.336, I=0.093, C=0.044 — one pass already gets you within 0.003 of every weight.

Which gives us:

```
w_D (Disability & illness) = 0.527 (carries most weight)
w_A (Age) = 0.336
w_I (Income) = 0.093
w_C (Overcrowding) = 0.044 (carries least weight)
```

These sum to 1. Disability carries more than eleven times the weight of overcrowding. And we never chose those numbers directly — we chose pairwise comparisons, and the eigenvector extracted the weights already hiding inside them.

The Honesty Test

AHP does something that gut-feeling ranking *never* can: it checks whether your own judgements are internally consistent. Saaty measured this with the Consistency Index: C.I. = $(\lambda_{max} - n) / (n - 1)$, where n = 4. A perfectly consistent matrix gives $\lambda_{max} = n$ exactly and C.I. = 0. For our matrix, $\lambda_{max} \approx 4.096$, giving C.I. ≈ 0.032. Dividing by Saaty's random index for 4×4 matrices (R.I. = 0.90) yields a Consistency Ratio of C.R. ≈ 0.04 — comfortably below the accepted threshold of 0.10. The mathematics had audited our instincts and found them consistent.

The Score — and the Result That Surprised Me

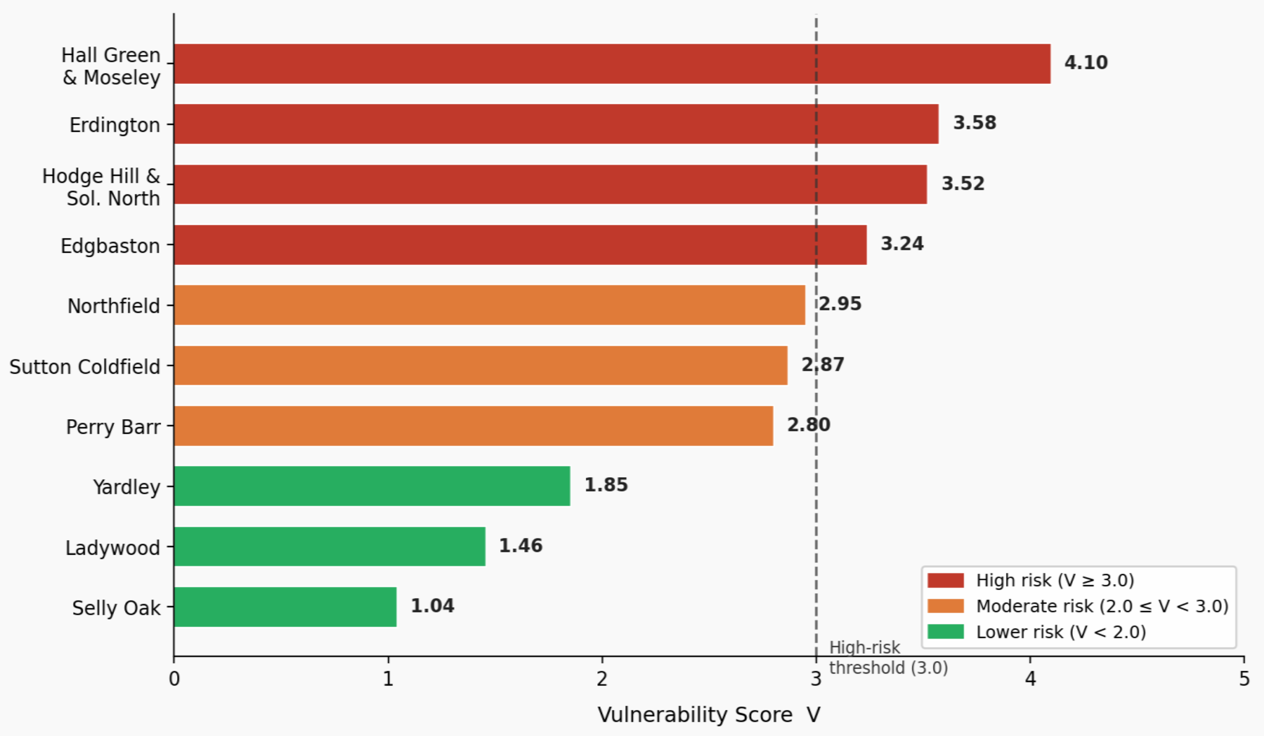
With the weights confirmed, vulnerability for each constituency is a weighted sum of four standardised factor scores, each rated 1–5 based on how far the constituency deviates from the Birmingham average:

```
V = 0.527·D + 0.336·A + 0.093·I + 0.044·C
```

Hall Green and Moseley ranked most vulnerable at V = 4.10, driven by the highest rate of serious illness in the city. Selly Oak ranked lowest at 1.04. Fine. Expected, almost.

Then I got to Ladywood. V = 1.46. Ninth out of ten — almost the least vulnerable constituency in Birmingham. I checked it three times.[3] Ladywood is widely regarded as one of Birmingham's most economically deprived areas. By income alone it looks like a priority. But the eigenvector gave income a weight of just 0.093, while disability carried 0.527. Ladywood's population skews younger and has below-average rates of serious illness. The mathematics did not confirm the intuition. It overruled it. And that overruling could redirect an ambulance from the ninth most vulnerable area to the first.

Figure 4 — Heatwave vulnerability scores across Birmingham constituencies. Ladywood ranks 9th despite being widely perceived as the most deprived area.



Why This Is More Than a Formula

AHP is in use everywhere humans must choose between incommensurable things: hospital triage, flood defence spending, corporate strategy, energy policy. The mathematics is always the same — a reciprocal matrix, an eigenvector, a consistency check. What makes it worth understanding is not the formula but the shift it forces.

You cannot hide behind vague instinct. Every pairwise comparison must be written down, justified, and submitted to a mathematical audit. If you disagree with the result, you can point to exactly which matrix entry produced it and argue for a different value. That is accountability that gut feeling can never offer.

Saaty found, buried in a reciprocal matrix, a way to make human judgement more honest with itself. On a day when Birmingham's streets were cooler than its living rooms and a city had to decide who needed help most — that matters. More than I expected when I first ran the numbers.

— — —

[1] This is the thermal mass paradox: the same insulation that slows heat loss in winter slows heat escape in summer. A building designed for a northern European climate is, in a real physical sense, the wrong building for the climate it will increasingly inhabit.

[2] Why stop at 9? Saaty drew on psychologist George Miller's 1956 finding that humans can reliably distinguish at most 7 ± 2 levels of a single stimulus — the "magical number seven." The nine-point scale was not arbitrary; it was calibrated to the limits of human perception.

[3] The disability standardised score for Ladywood is 1 out of 5 — the lowest in the city — because its population is relatively young with a below-average rate of very bad health. With disability carrying a weight of 0.527, this single factor pulls the total score down dramatically regardless of income. The eigenvector was right.

References

Saaty, T.L. (1980) The Analytic Hierarchy Process. New York: McGraw-Hill.

Miller, G.A. (1956) "The magical number seven, plus or minus two." Psychological Review, 63(2), pp. 81–97.

Vaidya, O.S. and Kumar, S. (2006) "Analytic hierarchy process: an overview of applications." European Journal of Operational Research, 169(1), pp. 1–29.

Birmingham City Observatory (2021) Constituency Data: Health, Age, Income, Overcrowding. Available at: https://cityobservatory.birmingham.gov.uk (Accessed: 13 April 2026).

Office for National Statistics (2021) Census 2021: Age by Constituency — Birmingham. Newport: ONS.

Birmingham City Council (2024) Average Annual Earnings by Constituency 2024. Available at: https://www.birmingham.gov.uk/download/downloads/id/2867/average_earnings_2024.pdf (Accessed: 13 April 2026).

Met Office (2022) UK Temperature Records: 19 July 2022. Available at: https://www.metoffice.gov.uk/about-us/press-office/news/weather-and-climate/2022/record-temperatures-in-the-uk (Accessed: 13 April 2026).

Appendix — Python Code

The following code reproduces the eigenvector weights and consistency check described in this essay. Running it takes under a second and returns exactly the numbers shown above.

```
import numpy as np

# Pairwise comparison matrix  [D, A, I, C]
M = np.array([
    [1,     2,     6,     9   ],
    [1/2,   1,     5,     7   ],
    [1/6,   1/5,   1,     3   ],
    [1/9,   1/7,   1/3,   1   ],
])

# Principal eigenvector via iterative normalisation
def ahp_weights(M, iterations=200):
    w = np.ones(M.shape[0]) / M.shape[0]
    for _ in range(iterations):
        w_new = M @ w
        w_new /= w_new.sum()
        if np.allclose(w, w_new, atol=1e-12): break
        w = w_new
    return w_new

w = ahp_weights(M)
print(f"Weights: D={w[0]:.4f}  A={w[1]:.4f}  I={w[2]:.4f}  C={w[3]:.4f}")
# → D=0.5266  A=0.3359  I=0.0931  C=0.0444

# Consistency check
lam_max = (M @ w / w).mean()
CI = (lam_max - 4) / 3
CR = CI / 0.90          # Random index for n=4 is 0.90
print(f"CR = {CR:.4f}") # → 0.0354  (threshold: 0.10)
```

Matplotlib figure code (Figure 1 — temperature model):

```
import matplotlib.pyplot as plt

hours  = list(range(24))
indoor = [28.83,26.72,25.75,25.16,25.20,24.27,26.72,32.30,34.95,
          38.83,40.45,42.35,42.35,43.36,43.39,41.38,41.31,39.38,
          39.45,38.90,34.88,34.88,34.33,33.33]
outdoor= [24.3,22.8,21.5,21.0,20.8,20.5,22.0,26.5,29.8,33.2,
          35.1,37.0,37.5,38.4,39.2,38.0,37.5,36.0,34.8,33.5,
          31.0,29.5,28.2,27.0]

fig, ax = plt.subplots(figsize=(9, 4))
ax.plot(hours, outdoor, "--", color="#e07b39", label="Outdoor")
ax.plot(hours, indoor,  "-",  color="#c0392b", label="Indoor (modelled)")
ax.fill_between(hours, outdoor, indoor,
    where=[i>=o for i,o in zip(indoor,outdoor)],
    alpha=0.13, color="#c0392b", label="Trapped heat")
ax.axhline(40, color="#444", linestyle=":", linewidth=1.1)
ax.set_xlabel("Hour of day"); ax.set_ylabel("Temperature (°C)")
ax.legend(); plt.tight_layout(); plt.show()
```

Author note: The heat transfer model and vulnerability scoring described here were developed as part of the 14-hour MathWorks M3 Challenge 2025, which I worked on as part of a team. All figures generated by me using Python (matplotlib).