Sam Nolan

Software Developer / Researcher

Education

2023 **MicroMasters Data, Economics and Development Policy**, *MITx*, Online (audited)

2018–2021 **Bachelor of Computer Science (Dean's Scholar, Honours)**, *RMIT*, Melbourne

2017 **VCE**, *Belmont High School*, Belmont, ATAR of 98.85

Experience

2022-2023 **Quantified Uncertainty Research Institute (QURI)**, *Software developer/Researcher*

Developing Squiggle, an estimation language that treats probability distributions as first class citizens. I did mostly frontend development on the language to add features required to do my independent research. I considered myself mainly a user of Squiggle that would develop when neccesary.

2022 **Independent**, *Researcher*

Conducted several projects into the construction of high quality CEAs. Such as:

1. Running a Dimensional Analysis on GiveWell's GiveDirectly CEA, identifying two implicit assumptions made.
2. Developing a computer language called Pedant, that ran a dimensional analysis on CEAs to identify implicit assumptions.
3. Led a research effort to quantify uncertainty in all GiveWell CEAs, as a submission to the GiveWell change our mind contest.

All my research is on the Effective Altruism Forum under the name *Hazelfire*

2022-2023 **Effective Altruism Melbourne**, *Secretary, President*

2020-2021 **RMIT Programming Club**, *President*

Projects

Squiggle Typescript. An estimation language that treats probability distributions as first class citizens. Made for the purpose of helping decision making around uncertainty, particularly in longtermist funding.

Pedant Haskell. A language to statically check for assumptions in Cost-Effectiveness Analyses

Uncertainty in GiveWell CEAs Squiggle. A created a collection of uncertainty estimates of GiveWell top charities

May Haskell / Elm. An time based organiser I wrote for managing a large amount of obligations

18a Gidding St – North Geelong, Victoria – Australia
☎ *+61 (474) 736 293* • ✉ *samnolan555@gmail.com* • in *Sam Nolan*
Hazelfire

Languages

Haskell	Very proficient, particularly in backends to do with GraphQL
Javascript / Typescript	Very proficient, used in building Squiggle, a React/ts probability library I maintained
Python	Very proficient, main programming language for use in AWS Lambda
Node.js	Lambda functions, chat bots
C++	Specialise in threading, networking, interpreters and performance