

# Polars POC – normalize clinvar



<https://d3b.atlassian.net/browse/SJRA-1096>

# Agenda

- Why Polars?
- Chosen strategy
- Live Demo
- Feedback
- Discussion

# Why Polars?

- Modern, intuitive Python API
- Fast: Rust, multi-core, in-memory
- Lazy execution & streaming support
- Rapidly growing popularity

# Chosen Strategy

- read vcf file: `polars-bio.scan_vcf`
- lazy execution
- streaming
- output: `polars.sink_parquet`

# polars-bio

- Meant for polars
- Support lazy/eager, streaming/in-memory
- Handles cloud storage (S3)
- Supports other formats (GFF, BED, FASTA, etc.)
- Limitation: vcf sample metrics ignored
- Alternatives considered:
  - o biobear: similar, but appears unmaintained
  - o non-polars Python libraries exist, but may not support streaming or lazy execution as well

# Live demo

- Minikube setup
- Venv setup

# Feedback

- PRO: easy to learn, syntax closed to spark
- CON: lack of maturity
  - Streaming feature fragile

# Discussion

- Next steps?
- Datalake-lib conversion effort