PQuery: A Flask-based web application for effective visualisation and filtering of large-scale multi-sample genetics variant datasets

**Authors:** Dionysios Grigoriadis, Alan Pittman, Pia Ostergaard

**Background:** As sequencing of exomes and genomes becomes cheaper and more easily accessible, genetics labs accumulate loads of sequencing-derived genetic variant datasets either for diagnostic purposes or for identifying new disease-causing genome variations. Despite there are numerous well-established command-line tools enabling efficient data querying and filtering, there is a lack of open source and adjustable graphical applications with similar function. Thus, without prior programming knowledge, it is extremely challenging for researchers to efficiently access, query and filter the large variant datasets for conducting either family/individual-based or cohort-wide genetics studies.

**Aim:** PQuery was developed to tackle this issue and facilitate the visualisation and interpretation of multi-sample large genetics variant datasets with an easy to use interface. PQuery is a Flask-based web application enabling fast interactive visualisation, querying, flexible filtering, and alignment inspection of genetic variant data.

**Features:**

Without programming skills required, users can automatically convert their input VCF file (compressed or uncompressed, annotated or not annotated) to an indexed and structured SQLite database, fully adjusted to the imported VCF file. Once the database is created and stored, PQuery can open in a browser window and instantly run on it with no need for re-importing every time it runs.

The application allows users to query the genetic variants for specific samples and genomic regions and it then instantly returns a table with the queried variants (one variant per row) and all fields present in the original VCF file. It also dynamically calculates the allele counts and frequencies for the selected samples.

PQuery can run in two modes: a) the cohort mode in which no sample-specific information is included in the returned table and b) the sample-specific mode in which the generated table will fully include the sample names and sample-related information. The first would be faster and more convenient for a cohort-wide study while the latter would be more suitable for family/trio studies. Users can efficiently filter all columns in both modes by using a customisable filter pane and inspect the variant calls through the integrated IGV viewer.

PQuery is designed to be easy to install on any machine and can be used by anyone. As a Flask application, it can be directly executed by individual users or deployed on a local server. The fact that it is using SQLite for data management means that the database runs as part of the app and does not require a separate server.