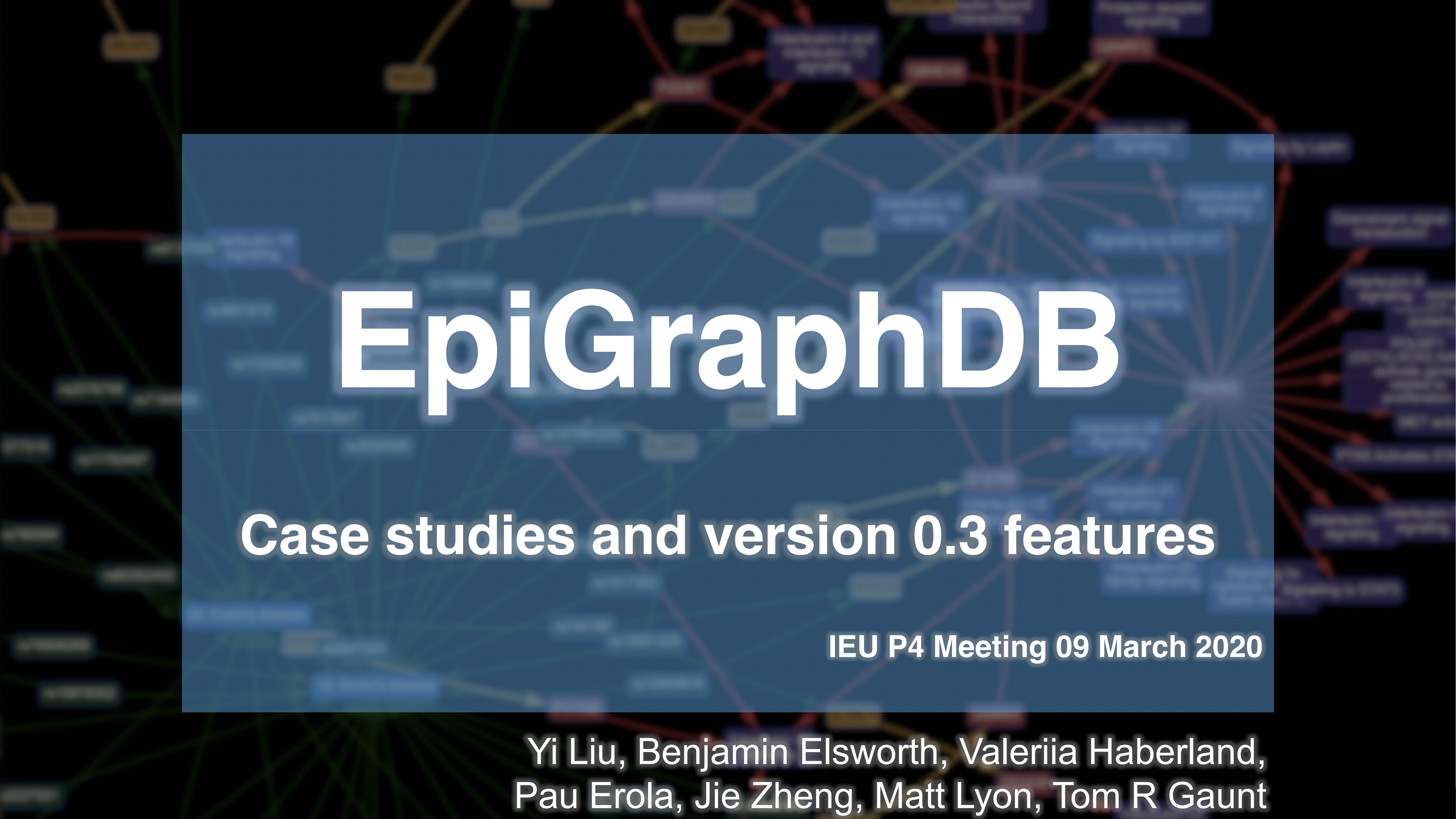


EpiGraphDB

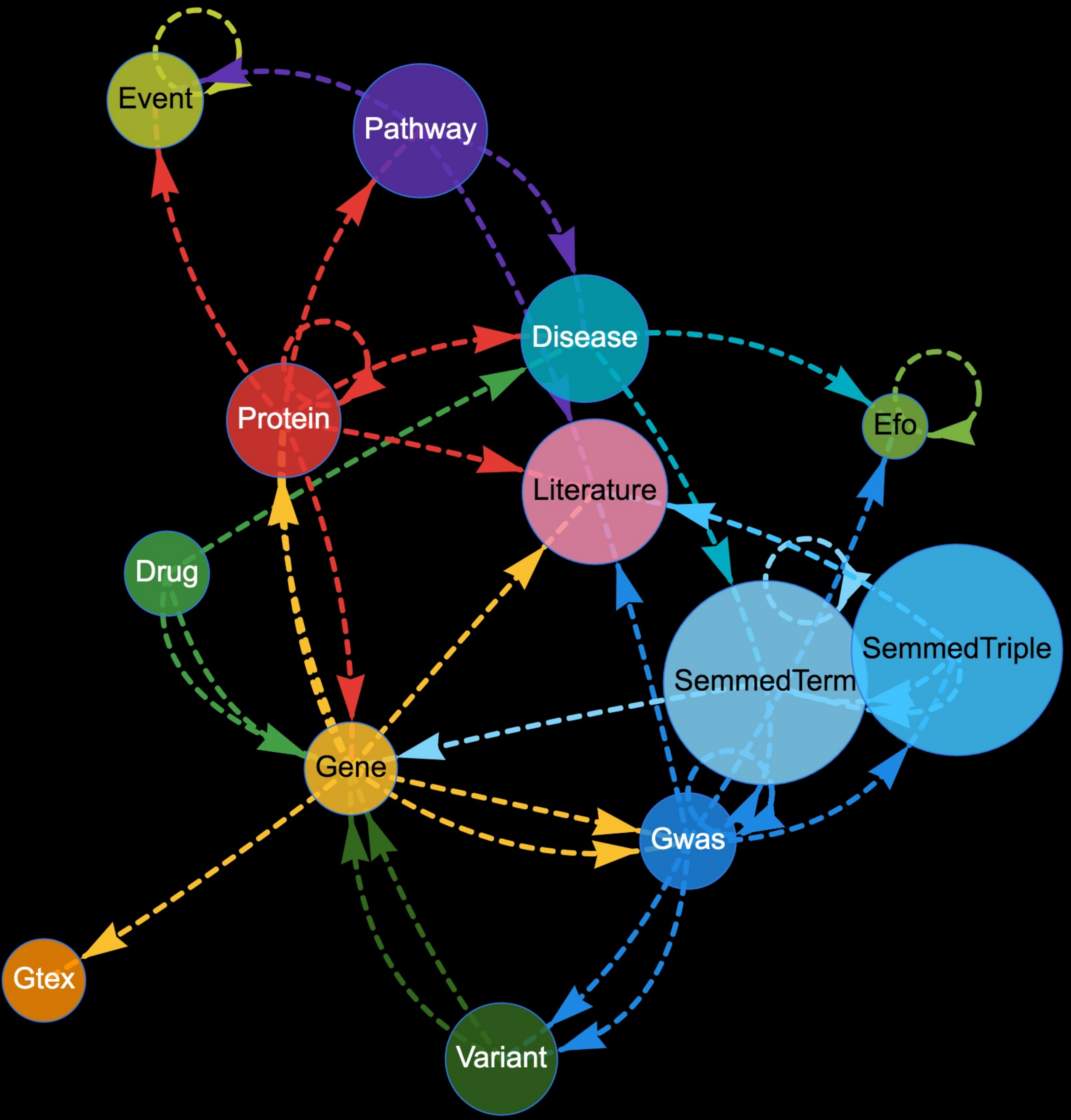
Case studies and version 0.3 features

IEU P4 Meeting 09 March 2020

Yi Liu, Benjamin Elsworth, Valeriia Haberland,
Pau Erola, Jie Zheng, Matt Lyon, Tom R Gaunt

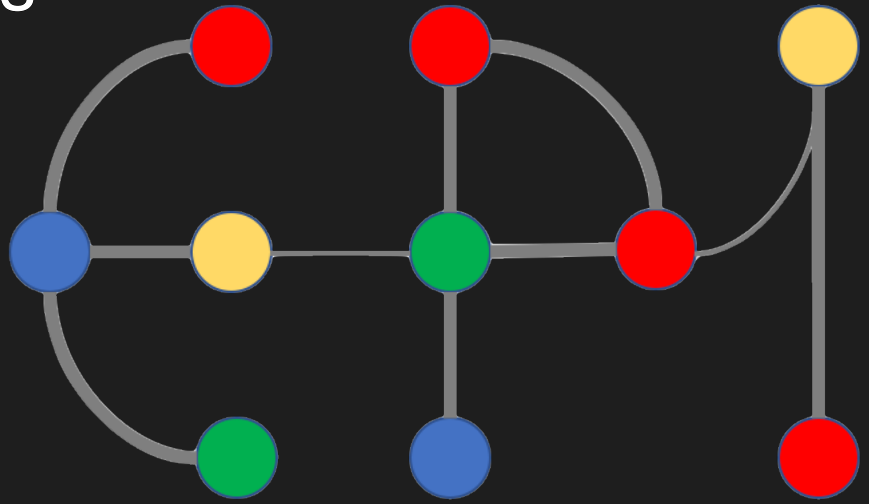
Outline

- Introduction
- EpiGraphDB version 0.3
- Use case 1: Pleiotropy
- Use case 2: Alternative drug targets
- Use case 3: Literature

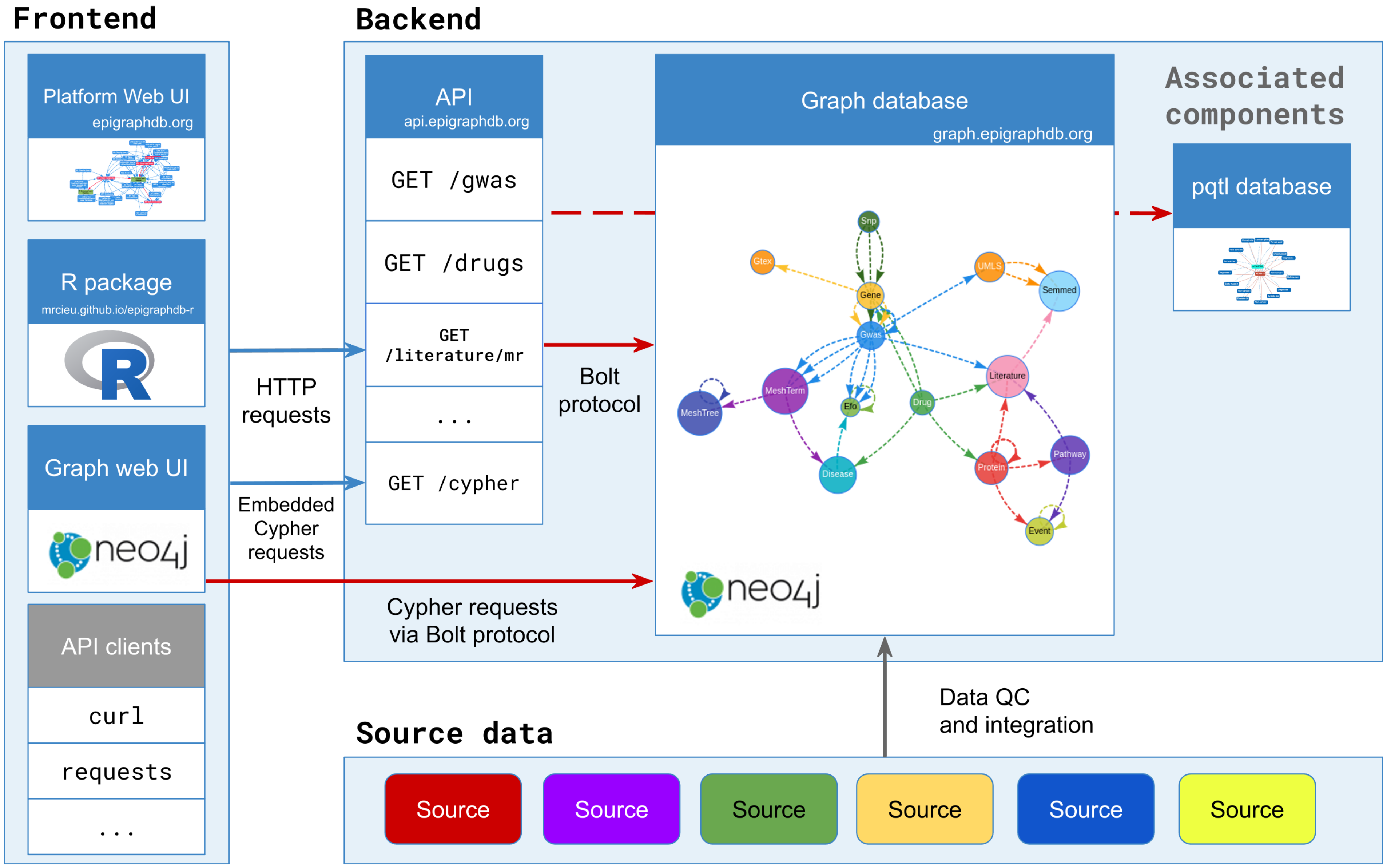


Integrated epidemiological evidence
http://docs.epigraphdb.org

- Causal relationships
- Association relationships
- Molecular pathways
- Literature mined / derived evidence
- Others



EpiGraphDB platform



Frontend

Platform Web UI
epigraphdb.org

R package
mrcieu.github.io/epigraphdb-r

Graph web UI

API clients

curl

requests

...

HTTP requests

Embedded Cypher requests

Backend

API
api.epigraphdb.org

GET /gwas

GET /drugs

GET /literature/mr

...

GET /cypher

Bolt protocol

Cypher requests via Bolt protocol

Graph database
graph.epigraphdb.org

Associated components

pqtl database

Data QC and integration

Source data

Source | Source | Source | Source | Source | Source

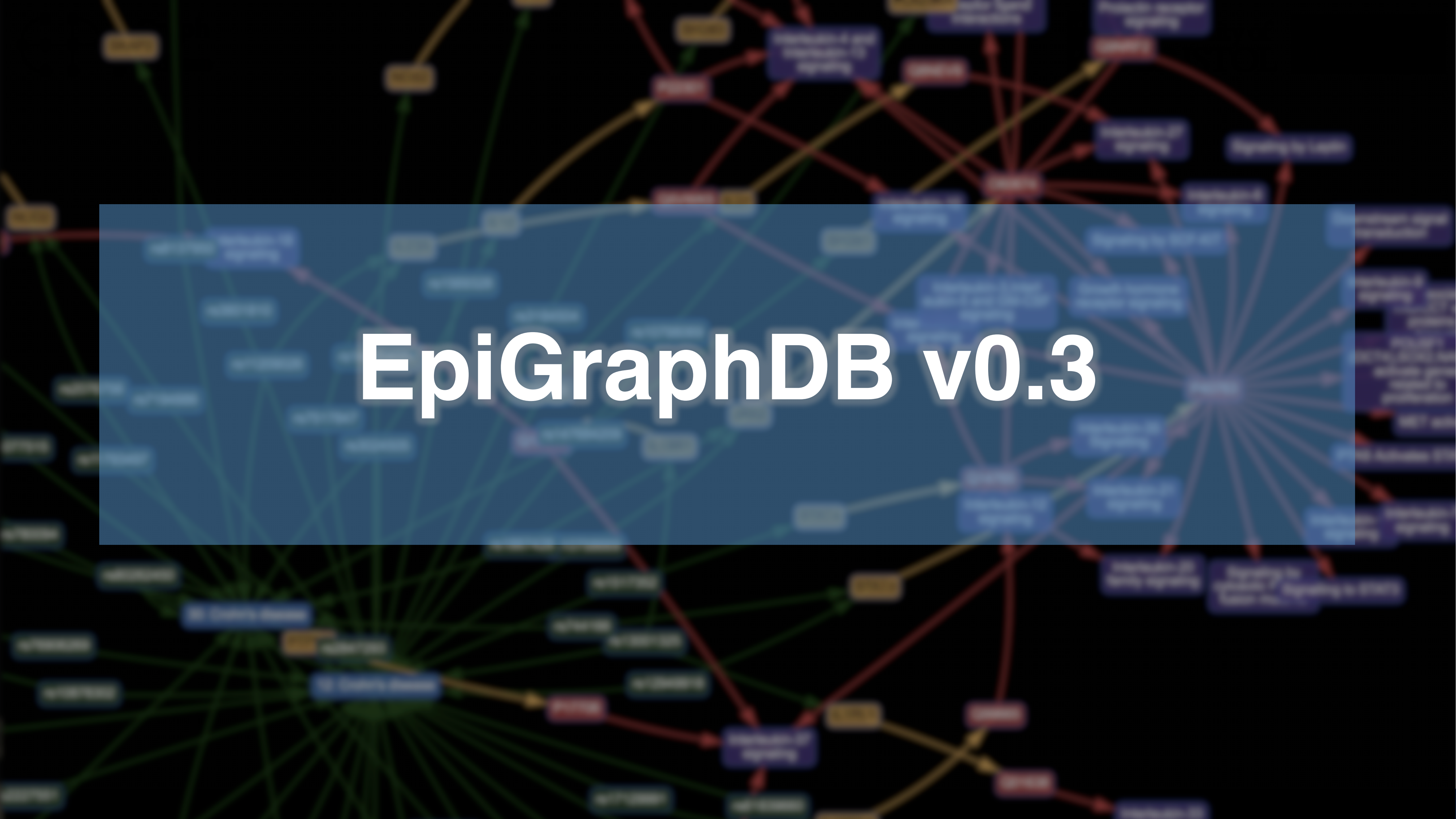
- Architected around a Neo4j graph database and queried via cypher
- Data integrated from various internal and external data sources
- Users can query EpiGraphDB from the API web service, and from the R client package
- A Web UI to showcase underlying evidence
- Working paper
- Companion Jupyter notebooks for case studies

IEU monthly meeting slides

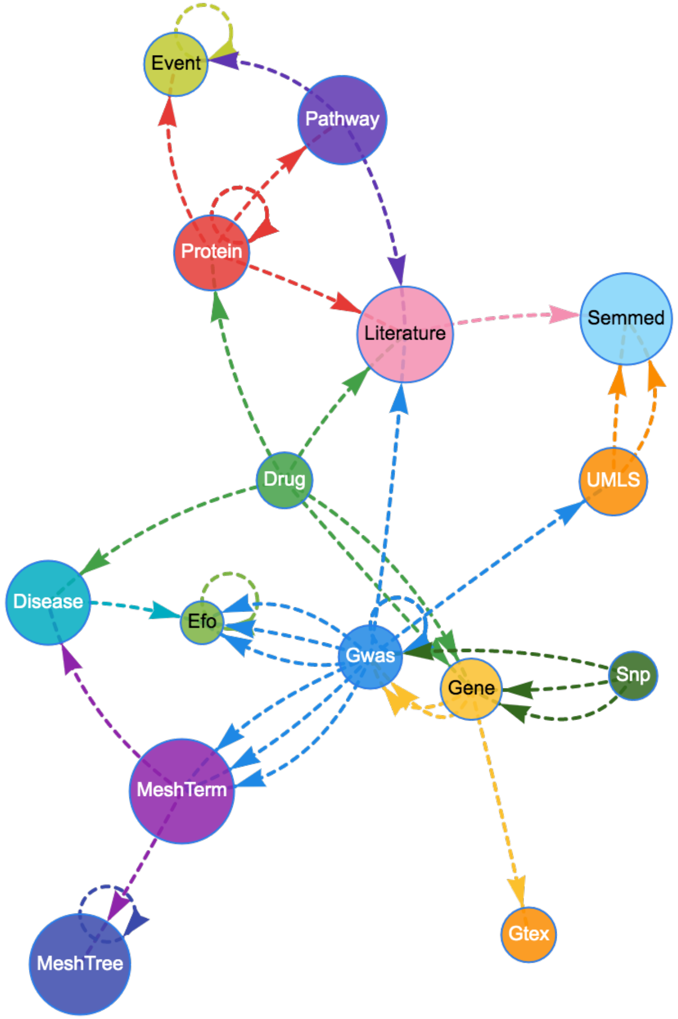
http://docs.epigraphdb.org/slides/2019-12-ieu-meeting.pdf

- Research project
- Case studies
- Platform
- Version 0.2

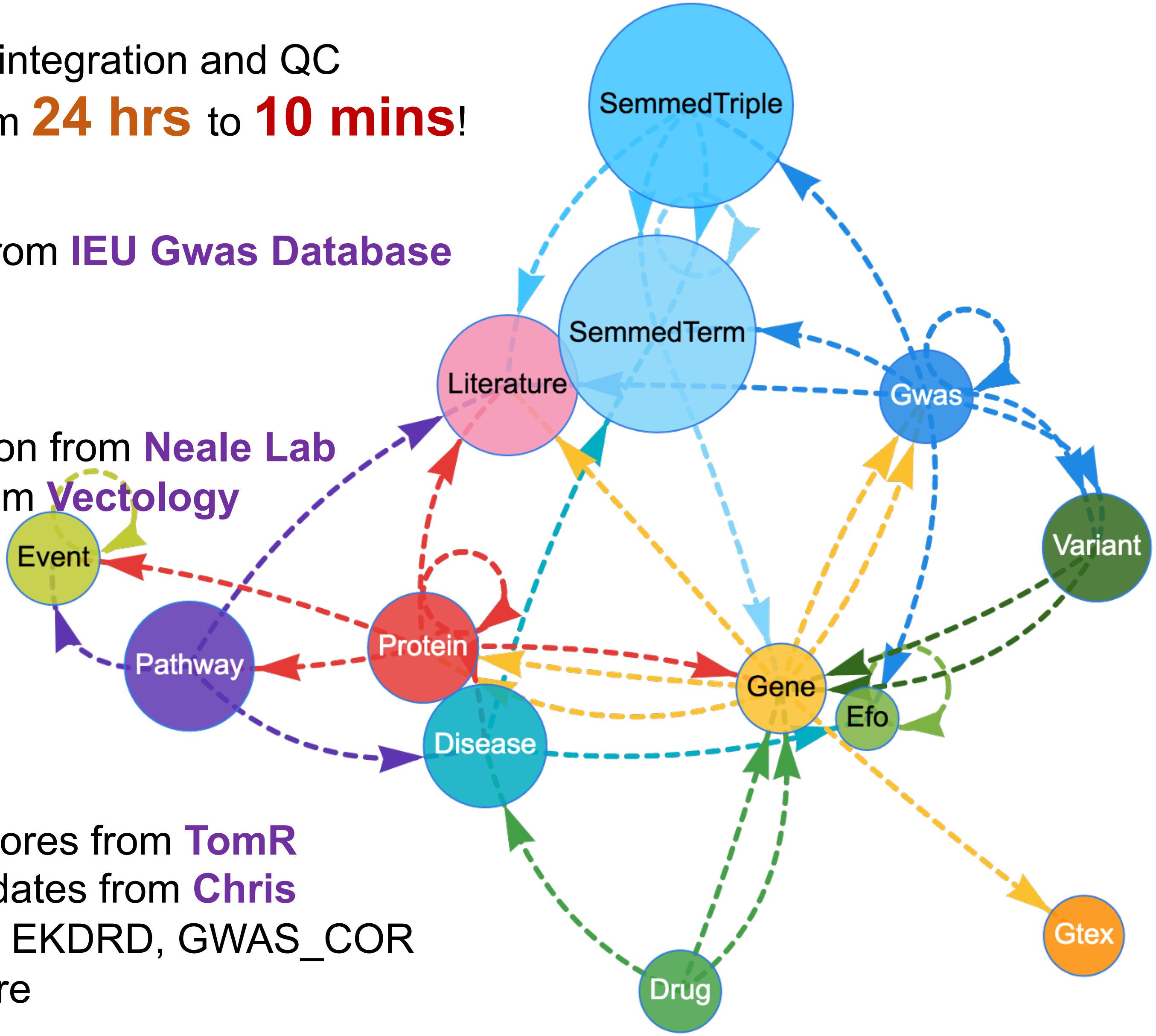
EpiGraphDB v0.3



A new graph database

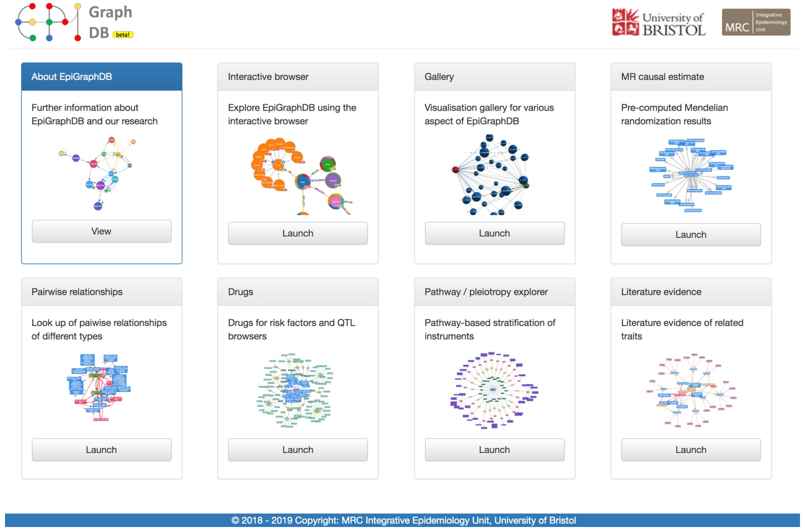


- Refactored data integration and QC
- Graph builds from **24 hrs** to **10 mins**!
- **10x** bigger!
- Updated Gwas from **IEU Gwas Database**
- SemMedDB
- MELODI
- EFO
- Genetic correlation from **Neale Lab**
- NLP mapping from **Vectology**



- Polygenic risk scores from **TomR**
- eQTL / pQTL updates from **Chris**
- Dropped: MeSH, EKDRD, GWAS_COR
- … and many more

A new Web UI



Graph DB beta!

About EpiGraphDB

Further information about EpiGraphDB and our research

View

Interactive browser

Explore EpiGraphDB using the interactive browser

Launch

Gallery

Visualisation gallery for various aspect of EpiGraphDB

Launch

MR causal estimate

Pre-computed Mendelian randomization results

Launch

Pairwise relationships

Look up of paiwise relationships of different types

Launch

Drugs

Drugs for risk factors and QTL browsers

Launch

Pathway / pleiotropy explorer

Pathway-based stratification of instruments

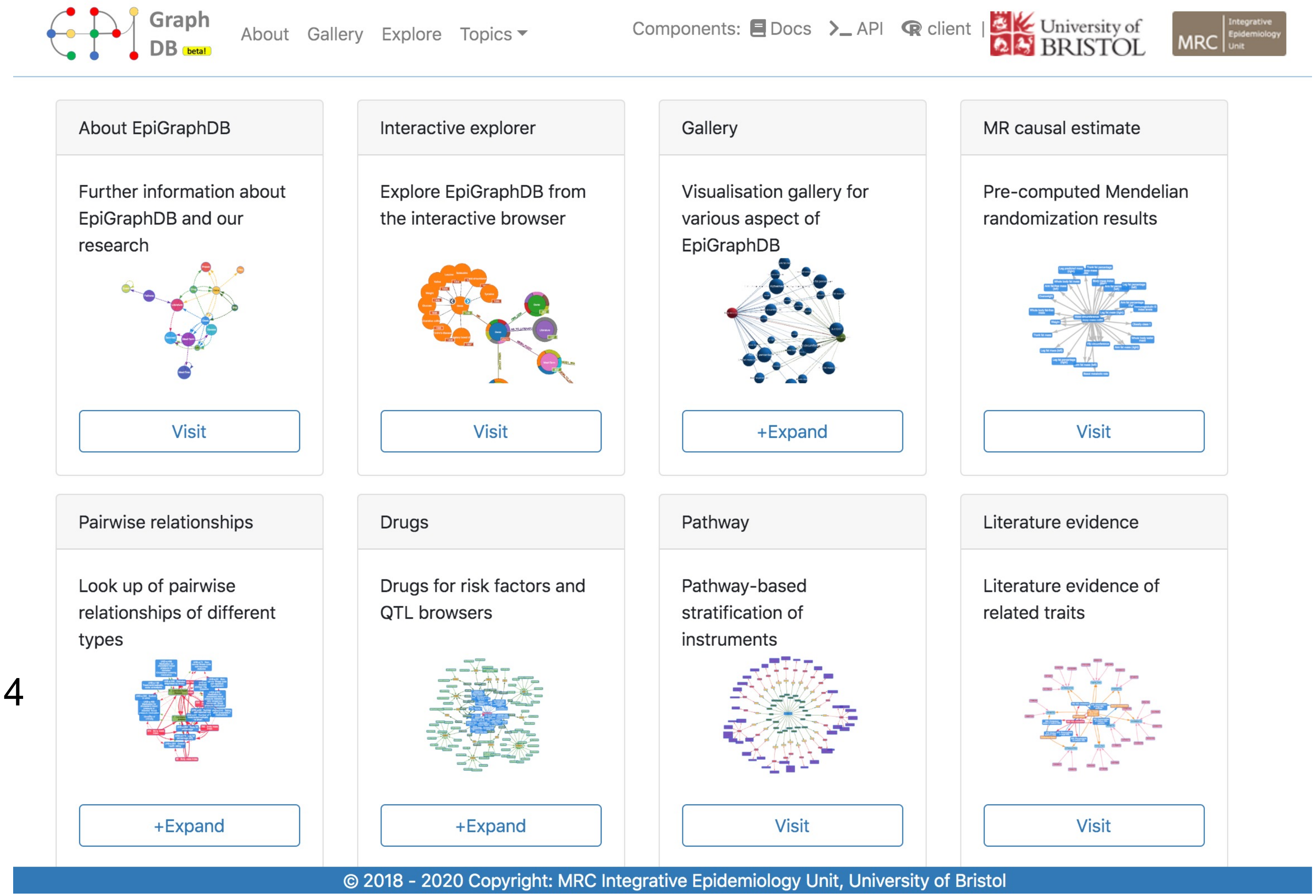
Launch

Literature evidence

Literature evidence of related traits

Launch

© 2018 - 2019 Copyright: MRC Integrative Epidemiology Unit, University of Bristol



Graph DB beta! About Gallery Explore Topics ▾ Components: Docs >_ API client |

About EpiGraphDB

Further information about EpiGraphDB and our research

Visit

Interactive explorer

Explore EpiGraphDB from the interactive browser

Visit

Gallery

Visualisation gallery for various aspect of EpiGraphDB

+Expand

MR causal estimate

Pre-computed Mendelian randomization results

Visit

Pairwise relationships

Look up of pairwise relationships of different types

+Expand

Drugs

Drugs for risk factors and QTL browsers

+Expand

Pathway

Pathway-based stratification of instruments

Visit

Literature evidence

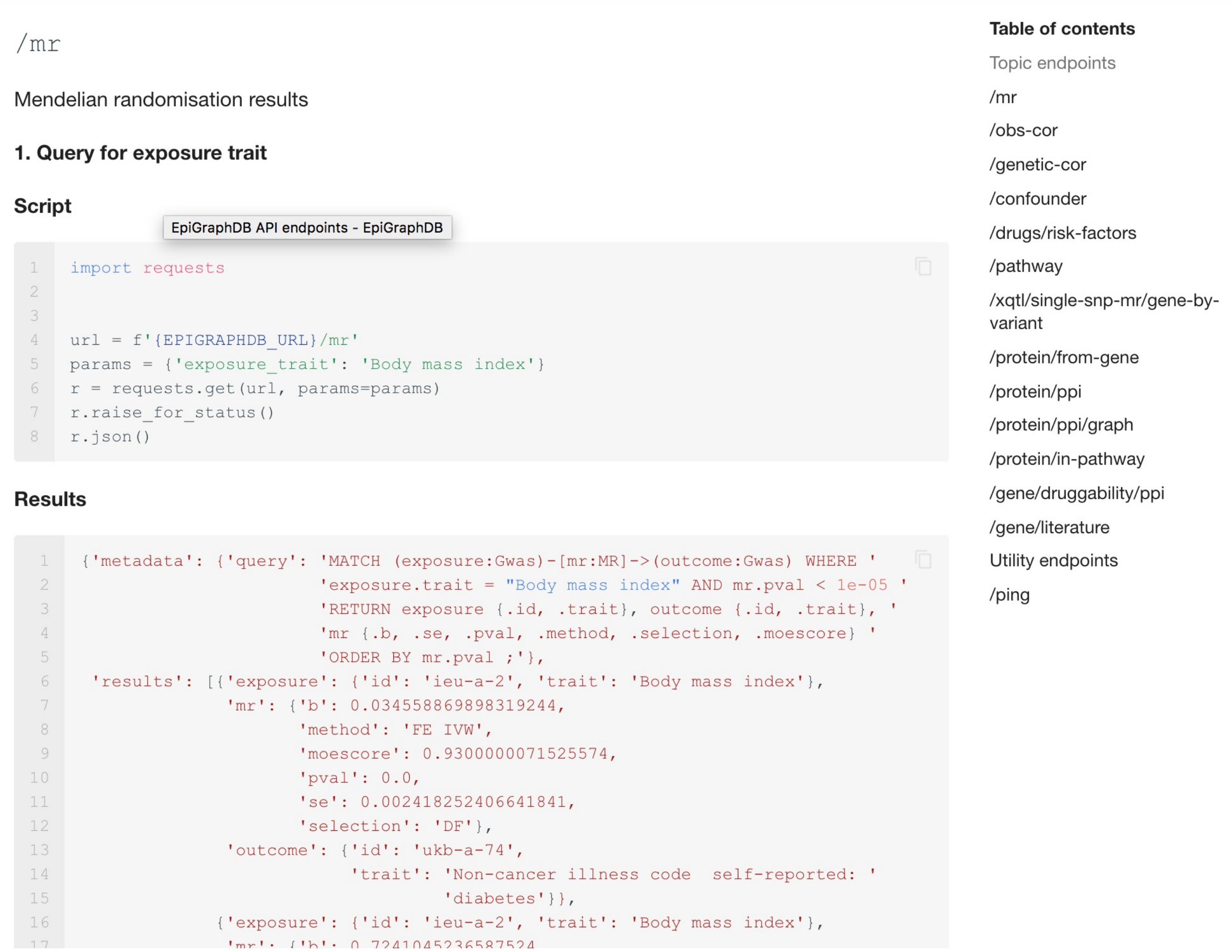
Literature evidence of related traits

Visit

© 2018 - 2020 Copyright: MRC Integrative Epidemiology Unit, University of Bristol

- v0.2: Flask + Jinja + Bootstrap3
- v0.3: FastAPI + Vue.js + Bootstrap 4
- Improved event handling
- Improved mobile support
- Improved caching

Improvements to API



/mr

Mendelian randomisation results

1. Query for exposure trait

Script

EpiGraphDB API endpoints - EpiGraphDB

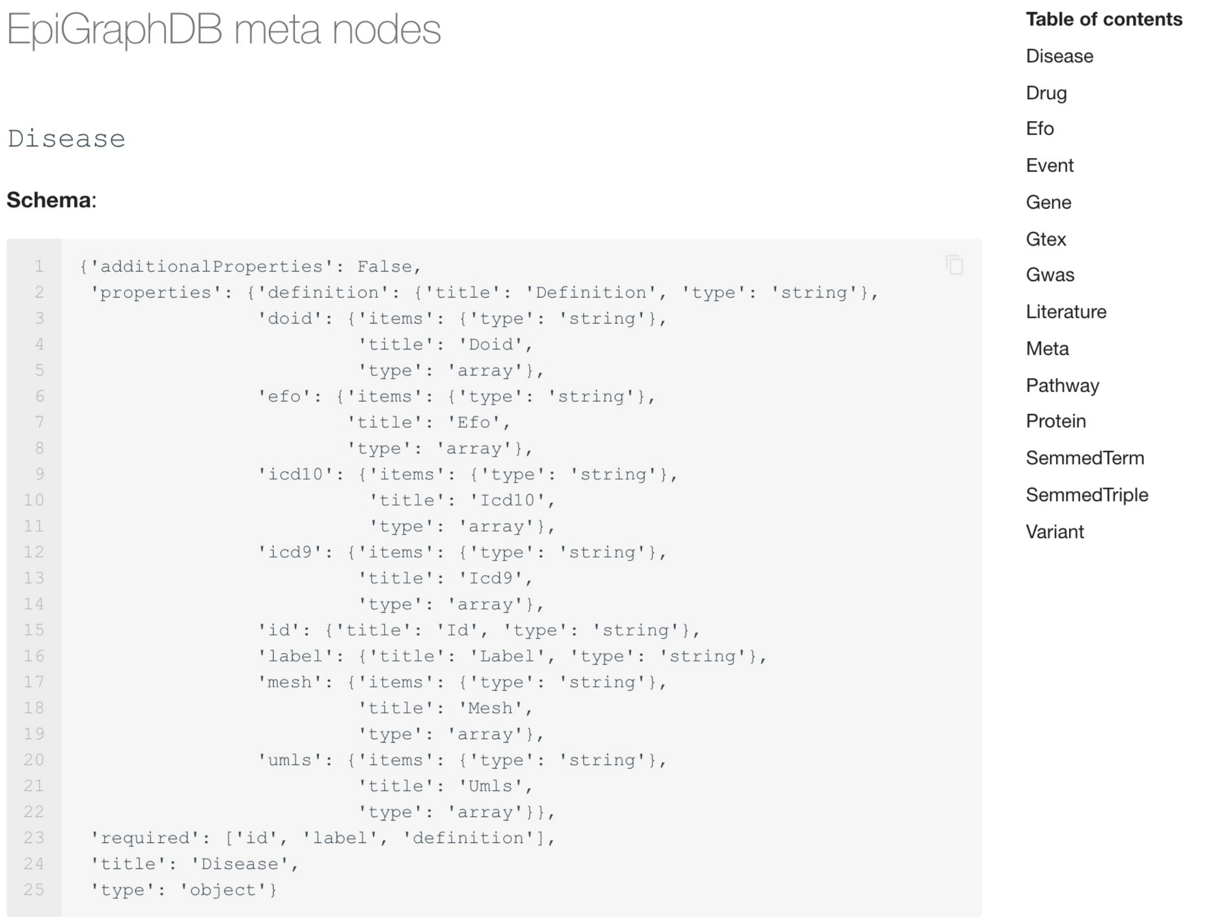
```
import requests


url = f'{EPIGRAPHDB_URL}/mr'
params = {'exposure_trait': 'Body mass index'}
r = requests.get(url, params=params)
r.raise_for_status()
r.json()
```

Results

```
{'metadata': {'query': 'MATCH (exposure:Gwas)-[mr:MR]->(outcome:Gwas) WHERE '
                        'exposure.trait = "Body mass index" AND mr.pval < 1e-05 '
                        'RETURN exposure {.id, .trait}, outcome {.id, .trait}, '
                        'mr {.b, .se, .pval, .method, .selection, .moescore} '
                        'ORDER BY mr.pval ;'},
 'results': [{'exposure': {'id': 'ieu-a-2', 'trait': 'Body mass index'},
              'mr': {'b': 0.034558869898319244,
                     'method': 'FE IVW',
                     'moescore': 0.9300000071525574,
                     'pval': 0.0,
                     'se': 0.002418252406641841,
                     'selection': 'DF'},
              'outcome': {'id': 'ukb-a-74',
                          'trait': 'Non-cancer illness code  self-reported: '
                                   'diabetes'}},
             {'exposure': {'id': 'ieu-a-2', 'trait': 'Body mass index'},
              'mr': {'b': 0.7241045236587524,
```

Table of contents

Topic endpoints
/mr
/obs-cor
/genetic-cor
/confounder
/drugs/risk-factors
/pathway
/xqtl/single-snp-mr/gene-by-variant
/protein/from-gene
/protein/ppi
/protein/ppi/graph
/protein/in-pathway
/gene/druggability/ppi
/gene/literature
Utility endpoints
/ping



EpiGraphDB meta nodes

Disease

Schema:

```
{'additionalProperties': False,
 'properties': {'definition': {'title': 'Definition', 'type': 'string'},
                'doid': {'items': {'type': 'string'},
                         'title': 'Doid',
                         'type': 'array'},
                'efo': {'items': {'type': 'string'},
                        'title': 'Efo',
                        'type': 'array'},
                'icd10': {'items': {'type': 'string'},
                          'title': 'Icd10',
                          'type': 'array'},
                'icd9': {'items': {'type': 'string'},
                         'title': 'Icd9',
                         'type': 'array'},
                'id': {'title': 'Id', 'type': 'string'},
                'label': {'title': 'Label', 'type': 'string'},
                'mesh': {'items': {'type': 'string'},
                         'title': 'Mesh',
                         'type': 'array'},
                'umls': {'items': {'type': 'string'},
                         'title': 'Umls',
                         'type': 'array'}},
 'required': ['id', 'label', 'definition'],
 'title': 'Disease',
 'type': 'object'}
```

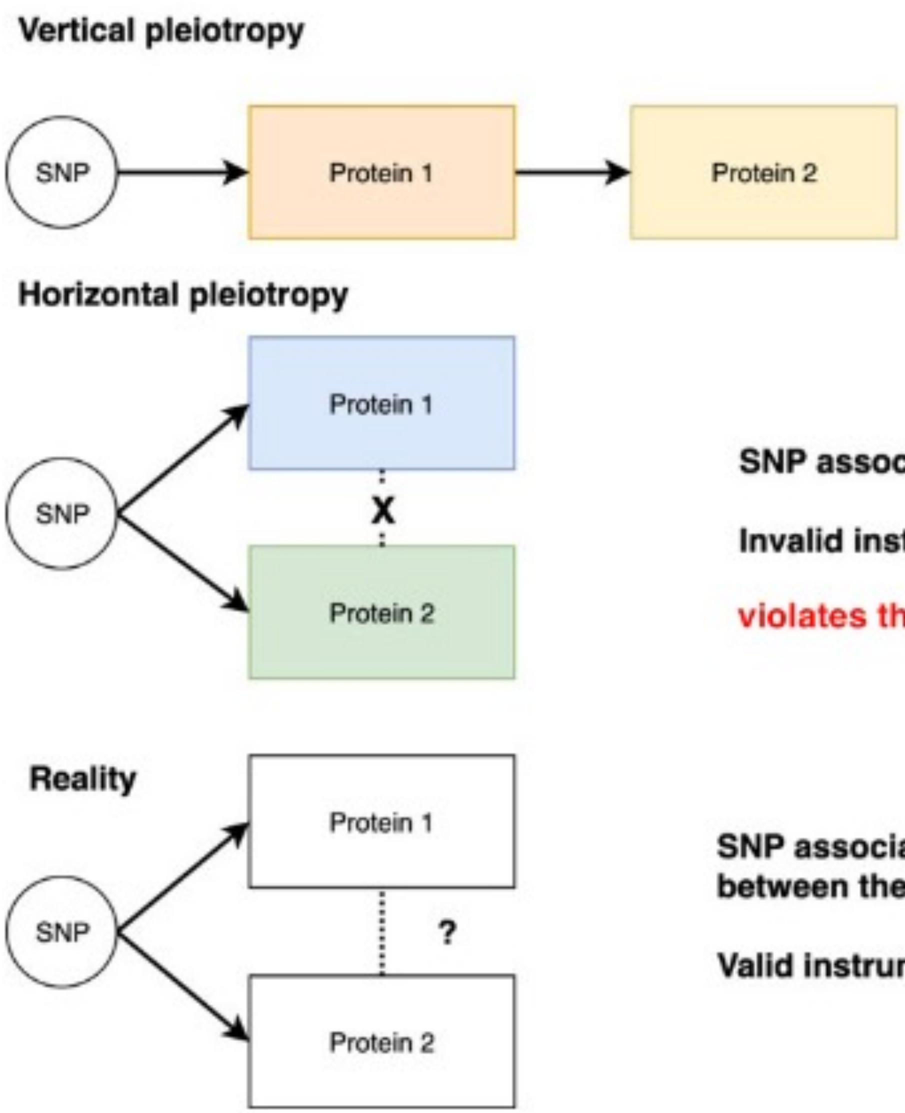
Table of contents

Disease
Drug
Efo
Event
Gene
Gtex
Gwas
Literature
Meta
Pathway
Protein
SemmedTerm
SemmedTriple
Variant

- Still working in progress

Case 1

Case 1 (1)



SNP affect proteins on the same pathway

Valid instrument for MR

SNP associated with proteins from different pathways

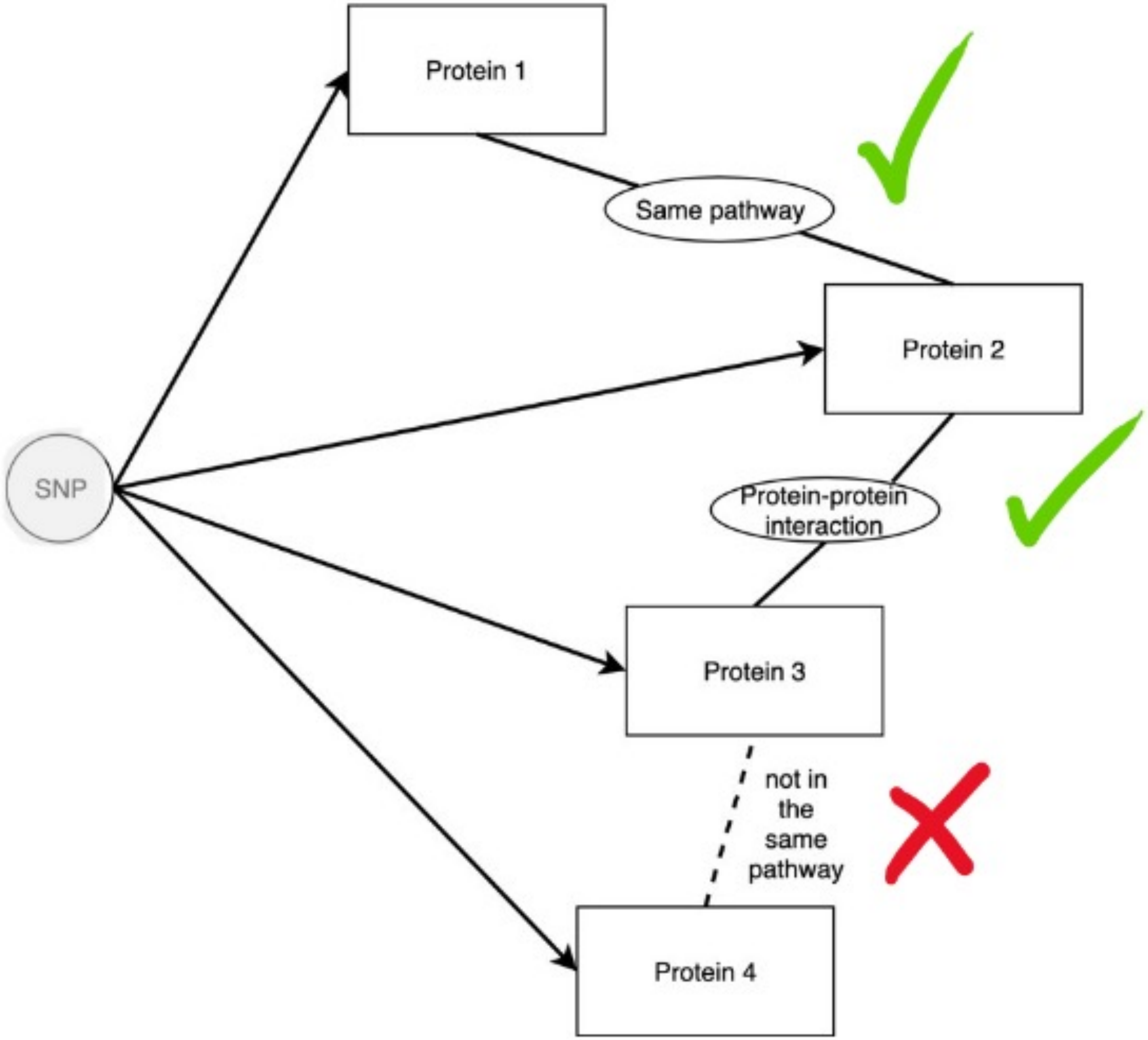
Invalid instrument for MR

violates the “exclusion restriction criterion”

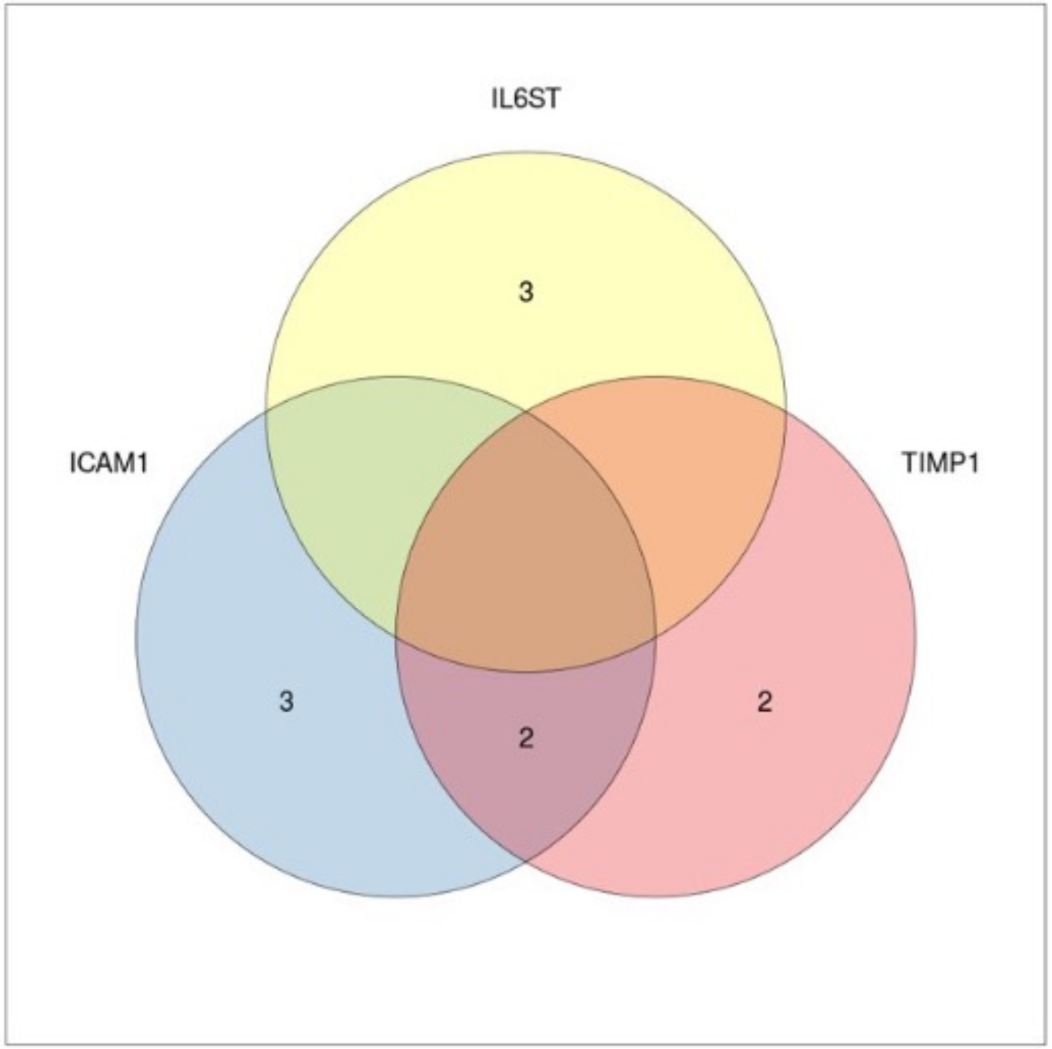
SNP associated with two proteins but relationship between the two proteins are missing

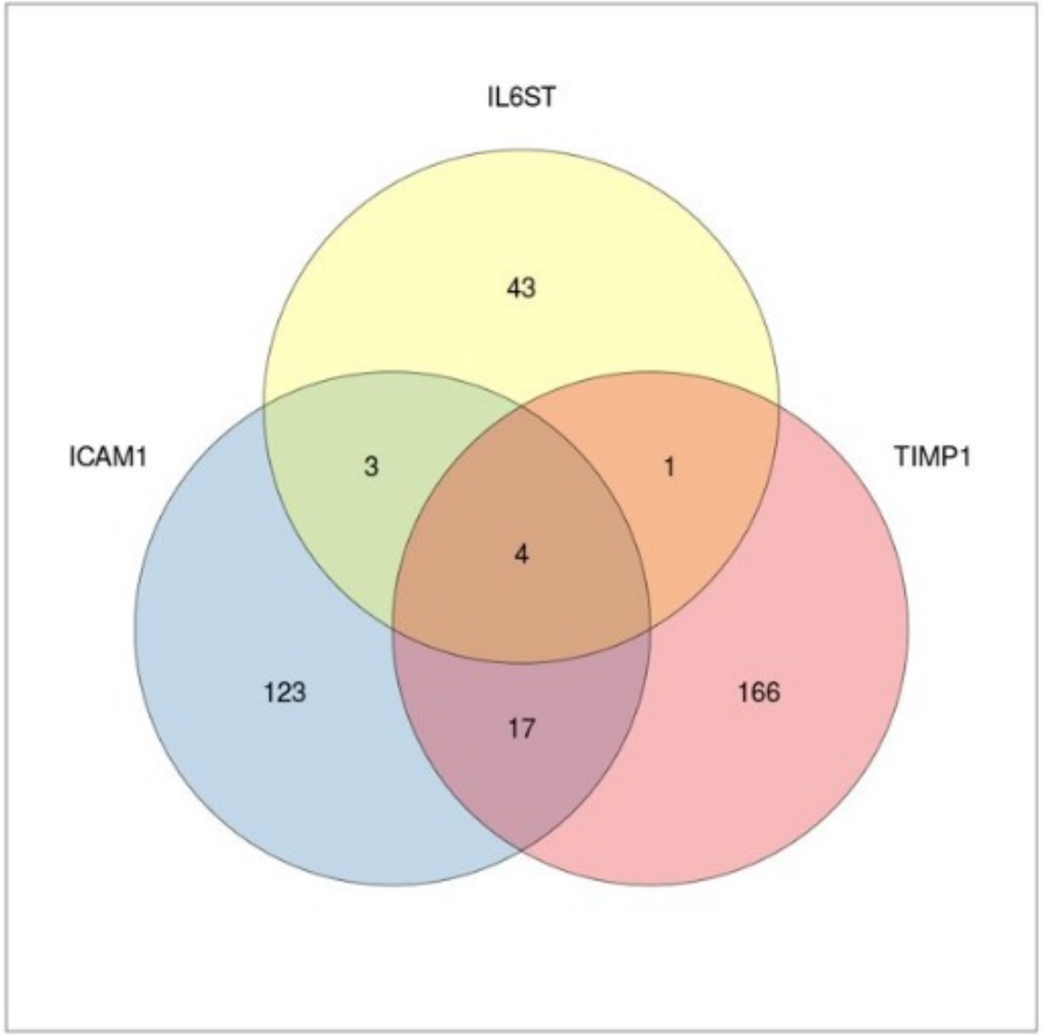
Valid instrument for MR?

Case 1 (2)


Protein 1
Same pathway
✓
Protein 2
SNP
Protein-protein interaction
✓
Protein 3
not in the same pathway
✗
Protein 4

Case 1 (3)

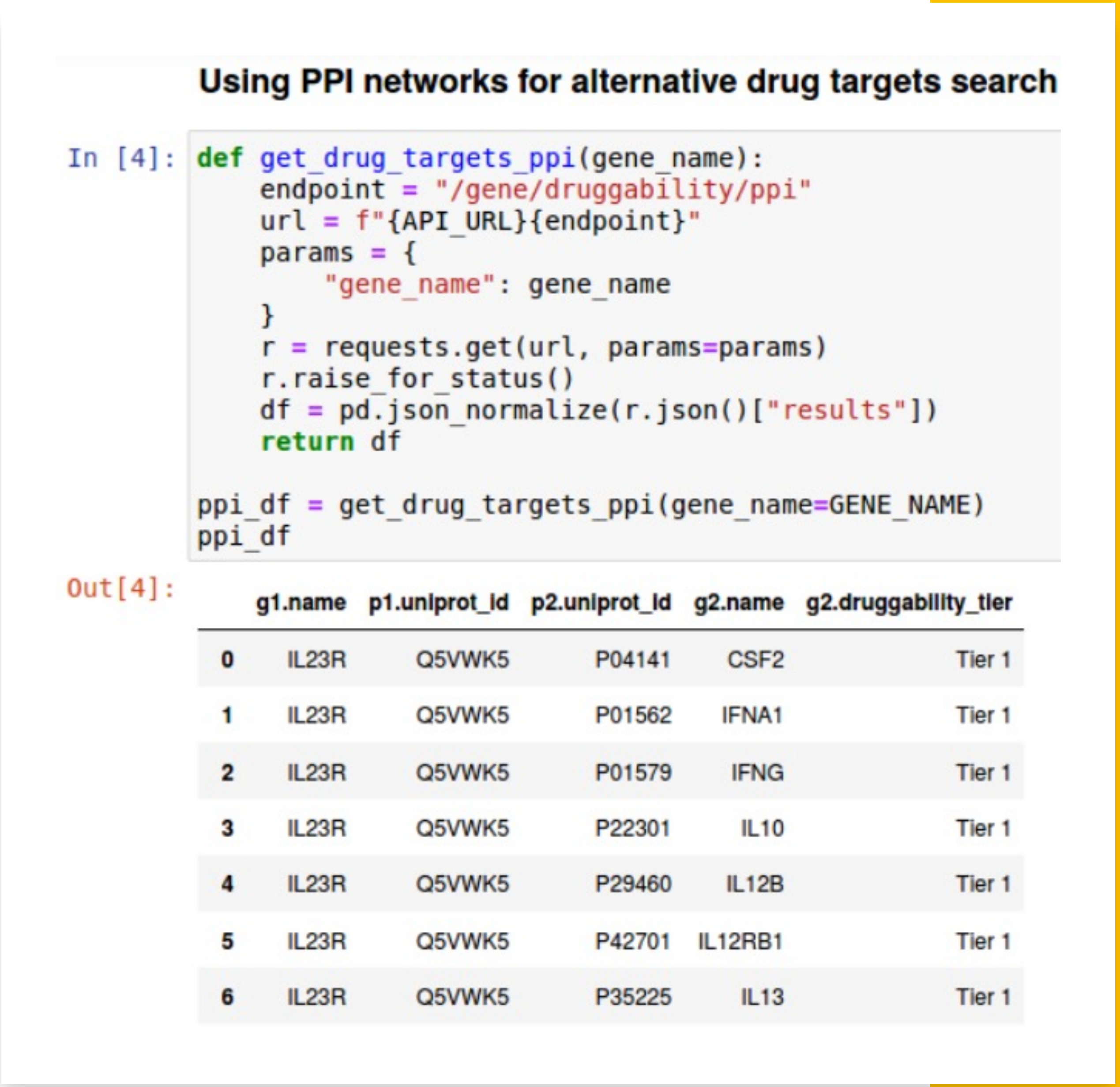




Case 2

IL23R and IBD

Search for interacting druggable proteins

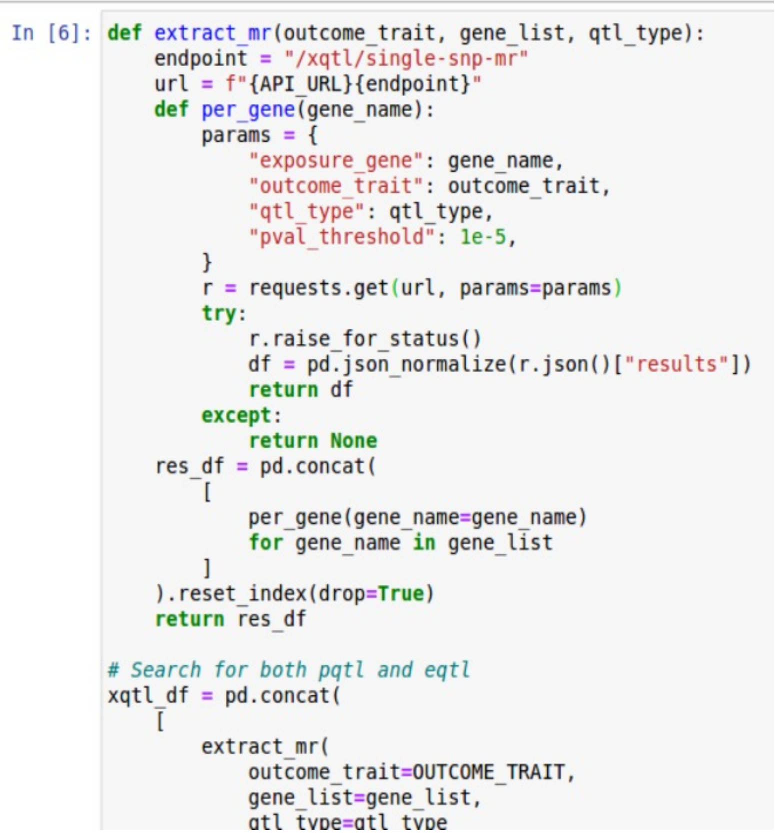


Using PPI networks for alternative drug targets search

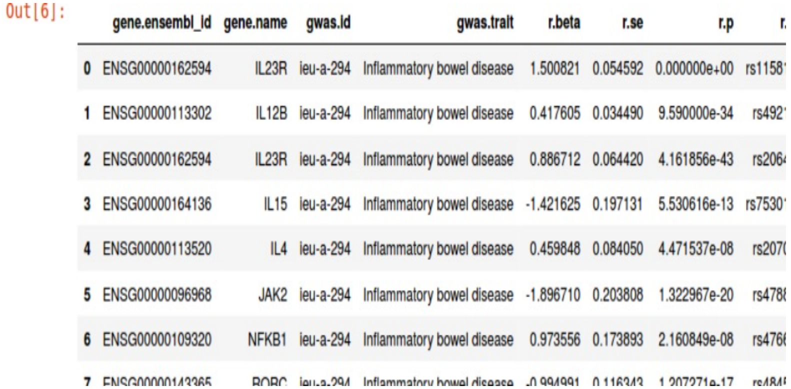
```
In [4]: def get_drug_targets_ppi(gene_name):
            endpoint = "/gene/druggability/ppi"
            url = f"{API_URL}{endpoint}"
            params = {
                "gene_name": gene_name
            }
            r = requests.get(url, params=params)
            r.raise_for_status()
            df = pd.json_normalize(r.json()["results"])
            return df

        ppi_df = get_drug_targets_ppi(gene_name=GENE_NAME)
        ppi_df
```

Out[4]:

	g1.name	p1.uniprot_id	p2.uniprot_id	g2.name	g2.druggability_tier
0	IL23R	Q5VWK5	P04141	CSF2	Tier 1
1	IL23R	Q5VWK5	P01562	IFNA1	Tier 1
2	IL23R	Q5VWK5	P01579	IFNG	Tier 1
3	IL23R	Q5VWK5	P22301	IL10	Tier 1
4	IL23R	Q5VWK5	P29460	IL12B	Tier 1
5	IL23R	Q5VWK5	P42701	IL12RB1	Tier 1
6	IL23R	Q5VWK5	P35225	IL13	Tier 1



```
In [6]: def extract_mr(outcome_trait, gene_list, qtl_type):
            endpoint = "/xqtl/single-snp-mr"
            url = f"{API_URL}{endpoint}"
            def per_gene(gene_name):
                params = {
                    "exposure_gene": gene_name,
                    "outcome_trait": outcome_trait,
                    "qtl_type": qtl_type,
                    "pval_threshold": 1e-5,
                }
                r = requests.get(url, params=params)
                try:
                    r.raise_for_status()
                    df = pd.json_normalize(r.json()["results"])
                    return df
                except:
                    return None
            res_df = pd.concat(
                [
                    per_gene(gene_name=gene_name)
                    for gene_name in gene_list
                ]
            ).reset_index(drop=True)
            return res_df

        # Search for both pqtl and eqtl
        xqtl_df = pd.concat(
            [
                extract_mr(
                    outcome_trait=OUTCOME_TRAIT,
                    gene_list=gene_list,
                    qtl_type=qtl_type
```

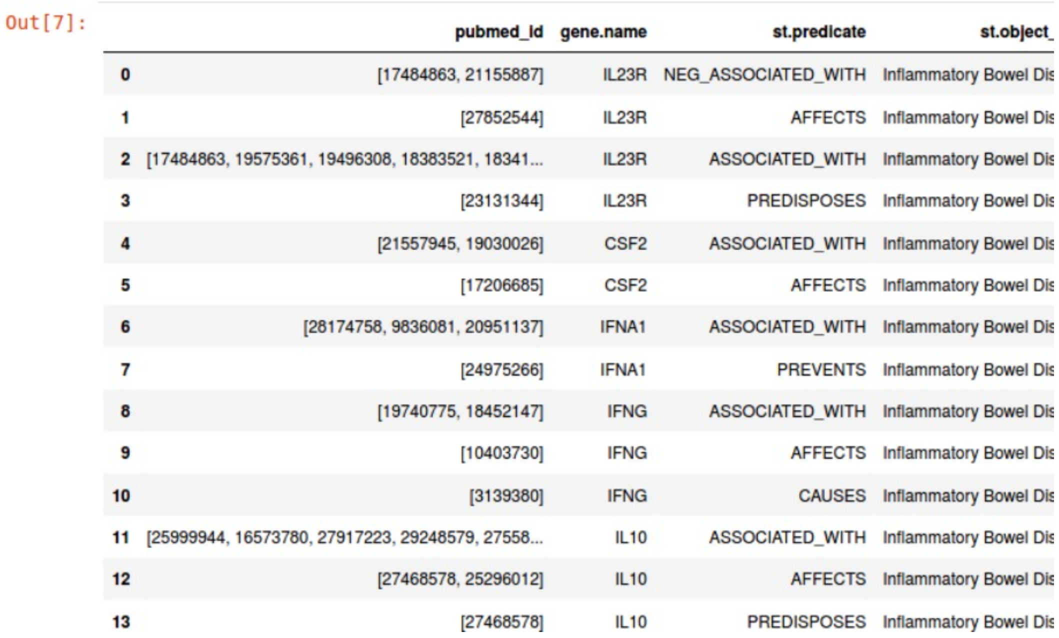


Out[6]:

	gene.ensembl_id	gene.name	gwas.id	gwas.trait	r.beta	r.se	r.p	r.
0	ENSG00000162594	IL23R	ieu-a-294	Inflammatory bowel disease	1.500821	0.054592	0.000000e+00	rs1158
1	ENSG00000113302	IL12B	ieu-a-294	Inflammatory bowel disease	0.417605	0.034490	9.590000e-34	rs492
2	ENSG00000162594	IL23R	ieu-a-294	Inflammatory bowel disease	0.886712	0.064420	4.161856e-43	rs206
3	ENSG00000164136	IL15	ieu-a-294	Inflammatory bowel disease	-1.421625	0.197131	5.530616e-13	rs7530
4	ENSG00000113520	IL4	ieu-a-294	Inflammatory bowel disease	0.459848	0.084050	4.471537e-08	rs207
5	ENSG00000096968	JAK2	ieu-a-294	Inflammatory bowel disease	-1.896710	0.203808	1.322967e-20	rs478
6	ENSG00000109320	NFKB1	ieu-a-294	Inflammatory bowel disease	0.973556	0.173893	2.160849e-08	rs476
7	ENSG00000143365	RORC	ieu-a-294	Inflammatory bowel disease	-0.994991	0.116343	1.207271e-17	rs484

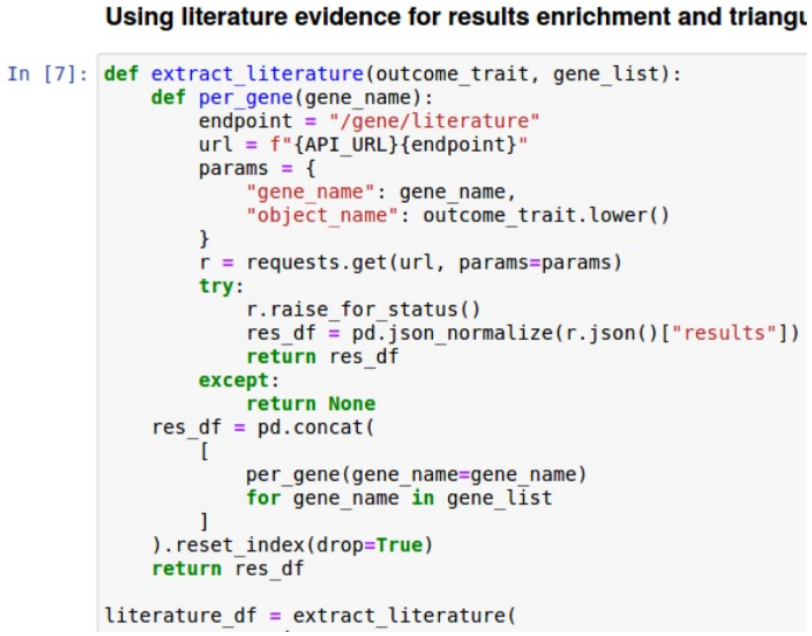
IL23R and IBD

Search for MR results for the Tier 1 interacting proteins



Out[7]:

	pubmed_id	gene.name	st.predicate	st.object_
0	[17484863, 21155887]	IL23R	NEG_ASSOCIATED_WITH	Inflammatory Bowel Dis
1	[27852544]	IL23R	AFFECTS	Inflammatory Bowel Dis
2	[17484863, 19575361, 19496308, 18383521, 18341...	IL23R	ASSOCIATED_WITH	Inflammatory Bowel Dis
3	[23131344]	IL23R	PREDISPOSES	Inflammatory Bowel Dis
4	[21557945, 19030026]	CSF2	ASSOCIATED_WITH	Inflammatory Bowel Dis
5	[17206685]	CSF2	AFFECTS	Inflammatory Bowel Dis
6	[28174758, 9836081, 20951137]	IFNA1	ASSOCIATED_WITH	Inflammatory Bowel Dis
7	[24975266]	IFNA1	PREVENTS	Inflammatory Bowel Dis
8	[19740775, 18452147]	IFNG	ASSOCIATED_WITH	Inflammatory Bowel Dis
9	[10403730]	IFNG	AFFECTS	Inflammatory Bowel Dis
10	[3139380]	IFNG	CAUSES	Inflammatory Bowel Dis
11	[25999944, 16573780, 27917223, 29248579, 27558...	IL10	ASSOCIATED_WITH	Inflammatory Bowel Dis
12	[27468578, 25296012]	IL10	AFFECTS	Inflammatory Bowel Dis
13	[27468578]	IL10	PREDISPOSES	Inflammatory Bowel Dis

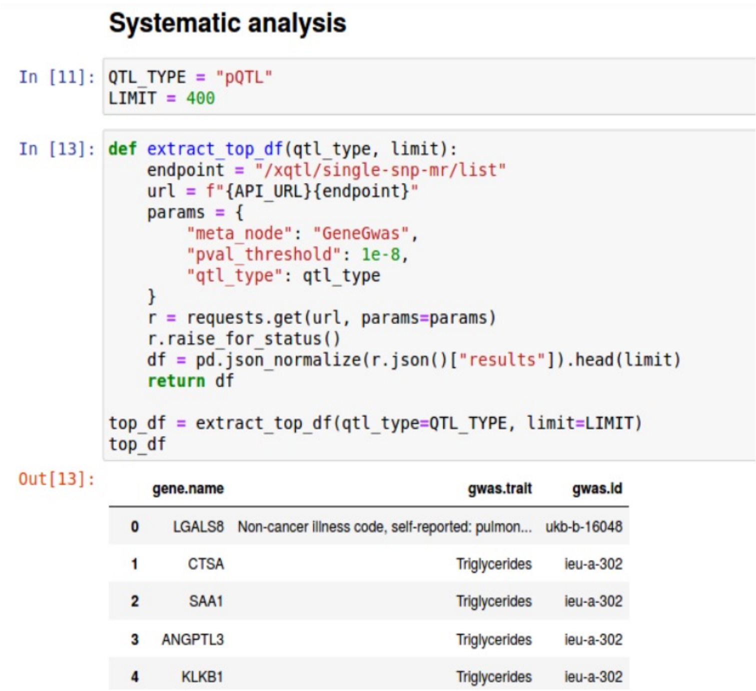


Using literature evidence for results enrichment and triangu

```
In [7]: def extract_literature(outcome_trait, gene_list):
            def per_gene(gene_name):
                endpoint = "/gene/literature"
                url = f"{API_URL}{endpoint}"
                params = {
                    "gene_name": gene_name,
                    "object_name": outcome_trait.lower()
                }
                r = requests.get(url, params=params)
                try:
                    r.raise_for_status()
                    res_df = pd.json_normalize(r.json()["results"])
                    return res_df
                except:
                    return None
            res_df = pd.concat(
                [
                    per_gene(gene_name=gene_name)
                    for gene_name in gene_list
                ]
            ).reset_index(drop=True)
            return res_df

        literature_df = extract_literature(
```

IL23R and IBD

Search for literature evidence of the interacting proteins



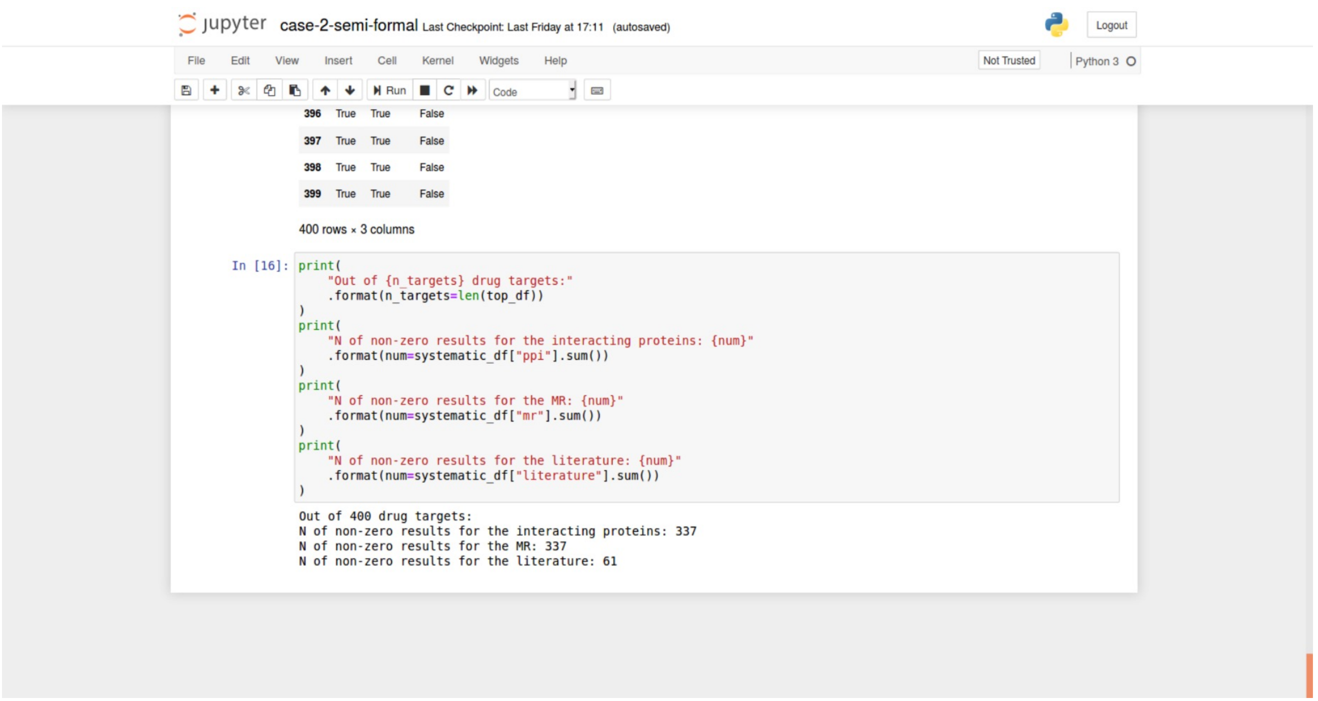
Systematic analysis

```
In [11]: QTL_TYPE = "pQTL"
         LIMIT = 400
```

```
In [13]: def extract_top_df(qtl_type, limit):
             endpoint = "/xqtl/single-snp-mr/list"
             url = f"{API_URL}{endpoint}"
             params = {
                 "meta_node": "GeneGwas",
                 "pval_threshold": 1e-8,
                 "qtl_type": qtl_type
             }
             r = requests.get(url, params=params)
             r.raise_for_status()
             df = pd.json_normalize(r.json()["results"]).head(limit)
             return df

         top_df = extract_top_df(qtl_type=QTL_TYPE, limit=LIMIT)
         top_df
```

Out[13]:

	gene.name	gwas.trait	gwas.id
0	LGALS8	Non-cancer illness code, self-reported: pulmon...	ukb-b-16048
1	CTSA	Triglycerides	ieu-a-302
2	SAA1	Triglycerides	ieu-a-302
3	ANGPTL3	Triglycerides	ieu-a-302
4	KLKB1	Triglycerides	ieu-a-302



396	True	True	False
397	True	True	False
398	True	True	False
399	True	True	False

400 rows × 3 columns

```
In [16]: print(
             "Out of {n_targets} drug targets:"
             .format(n_targets=len(top_df))
         )
         print(
             "N of non-zero results for the interacting proteins: {num}"
             .format(num=systematic_df["ppi"].sum())
         )
         print(
             "N of non-zero results for the MR: {num}"
             .format(num=systematic_df["mr"].sum())
         )
         print(
             "N of non-zero results for the literature: {num}"
             .format(num=systematic_df["literature"].sum())
         )
```

```
Out of 400 drug targets:
N of non-zero results for the interacting proteins: 337
N of non-zero results for the MR: 337
N of non-zero results for the literature: 61
```

Systematic analysis

Case 3

Case 3

Explore the literature evidence connecting two (or more) traits

1. Given an exposure, find disease traits with causal evidence
2. Select one (or more) exposure -> disease pair and extract literature evidence
3. Select subgraph of literature and extract publication information

Acknowledgements

EpiGraphDB

Yi Liu
Benjamin Elsworth
Valeriia Haberland
Pau Erola
Jie Zheng
Matt Lyon
Tom R Gaunt

pQTL project
Jie Zheng
Valeriia Haberland
Benjamin Elsworth
Denis Baird
Venexia Walker
Tom Richardson
Kurt Taylor
James Staley
George Davey Smith
Philip Haycock
Gibran Hemani
Robert Scott
Biogen & GSK
collaborators

MRC Integrative Epidemiology Unit

gsk GlaxoSmithKline

wellcome

Biogen

ORACLE CLOUD

University of BRISTOL

CANCER RESEARCH UK

docker

elasticsearch

FastAPI

Flask web development, one drop at a time

neo4j

jupyter

pandas

python

R

OPENAPI INITIATIVE

tidyverse

Reference