A Linked Data Representation for Summary Statistics and Grouping Criteria

James P. McCusker[1 [0000−0003−1085−6059]], Michel Dumontier[2 [0000−0003−4727−9435]], Shruthi Chari[1 [0000−0003−2946−7870]], Joanne S. Luciano[3 [0000−0002−1753−2885]], and Deborah L. McGuinness[1 [0000−0001−7037−4567]]

[1] Department of Computer Science, Rensselaer Polytechnic Institute. 110 8th Street, Troy, NY, USA {mccusj2,charis}@rpi.edu, dlm@cs.rpi.edu
[2] Maastricht University. Minderbroedersberg 4-6, 6211 LK Maastricht, Netherlands
[3] University of the Virgin Islands. Charlotte Amalie, St. Thomas, USVI, USA

Abstract. Summary statistics are fundamental to data science, and are the buidling blocks of statistical reasoning. Most of the data and statistics made available on government web sites are aggregate, however, until now, we have not had a suitable linked data representation available. We propose a way to express summary statistics across aggregate groups as linked data using Web Ontology Language (OWL) Class based sets, where members of the set contribute to the overall aggregate value. Additionally, many clinical studies in the biomedical field rely on demographic summaries of their study cohorts and the patients assigned to each arm. While most data query languages, including SPARQL, allow for computation of summary statistics, they do not provide a way to integrate those values back into the RDF graphs they were computed from. We represent this knowledge, that would otherwise be lost, through the use of OWL 2 punning semantics, the expression of aggregate grouping criteria as OWL classes with variables, and constructs from the Semanticscience Integrated Ontology (SIO), and the World Wide Web Consortium's provenance ontology, PROV-O, providing interoperable representations that are well supported across the web of Linked Data. We evaluate these semantics using a Resource Description Framework (RDF) representation of patient case information from the Genomic Data Commons, a data portal from the National Cancer Institute.

Keywords: Knowledge Representation · Linked Data · Provenance · Summary Statistics · Data Science · Transparency · Interoperability · Data Exploration.

1 Introduction

One of the most common forms of data analysis involves the use of basic statistics over groups of things. Sums, counts, and averages provide the foundation

for understanding data, and because of this, these aggregation functions form the basis of statistics and statistical analysis that underpins much of modern science. We provide a means to express aggregate facts about classes of entities in a way that avoids issues with the open world assumption by asserting those facts, relative to specific graphs, and then closing those graphs using cryptographic graph hash identities. This method can be used to automatically create classes through user interaction with semantically-aware, aggregation-based data exploration and analysis tools based on OnLine Analytical Processing (OLAP) [4] and statistical languages such as R [12].

This paper provides background on existing work to formalize aggregation criteria as OWL classes with URIs derived from those aggretation criteria using the RDF graph digest algorithm RGDA1 [17]. We extend it with a method for asserting aggregate facts about those OWL classes using the Semanticscience Integrated Ontology (SIO) [7] in order to provide support for realist representations of scientific data and knowledge [8]. These aggregate facts are treated as attributes of the class, grouping both different aggregate measures (like count, mean, and standard deviation) to specific kinds of attributes (like age and survival). Identifiers for the source RDF graph using the graph digest algorithm RGDA1 [17] can, if used in the provenance of those aggregate facts, will provide a closed set of assertions that the aggregate facts were computed over. We therefore provide a formal method to express the semantics of aggregate functions over well-defined grouping criteria. This allows us to create summary graphs of data that can be used in multiple contexts and introspected using reusable software. These mappings are illustrated using patient case information from the Genomic Data Commons (GDC) [11], a National Cancer Institute data portal for reusable cancer data.

2 Example Data Using Semanticscience Integrated Ontology

The expression of summary statistics requires a vocabulary that includes support for attributes, objects, and their interrelationships. As an integrated ontology, the Semanticscience Integrated Ontology (SIO) ontology includes terminology for reified roles, attributes (including quantities and qualities), time, and processes. SIO is an integrated ontology - it provides both an upper level framework for semantics and detailed semantics for general science. SIO is often extended for particular domains, and has been used effectively, for example, for modeling scientific data using relationships between entities and attributes. The top-level class of *sio:entity*[4] is subclassed by *sio:object* (continuents), *sio:attribute* (characteristics of any entity), and *sio:process* (occurrents). All kinds of entities can have attributes. The subtype of an attribute determines which attribute is being characterized, and *sio:'has value'* relates literal forms of the value of the attribute to it. Entities are related to their attributes using *sio:'has attribute'*.

[4] SIO properties and classes are expressed here using their labels, and quoted when necessary.

We use case data for our examples from the Genomic Data Commons (GDC) [11], that includes patient demographics, diagnosis, and survival. This kind of data is representative of biomedical information and includes information attributes with and without temporality, roles between entities, and some processes. GDC is particularly interesting because much of the exploration of data within the GDC is based on displaying aggregate statistics, even though there is no way to use those statistics outside of the user interfaces they provide. Our example can set us on a path to pre-computing aggregate statistics and providing not just visualizations of them, but also provide generalized API access to that information in new ways. While GDC does not provide examples of all possible RDF-based data, it is sufficient to illustrate our summary statistics approach. We also discuss below how to apply these approaches to all possible RDF by providing reification rules for both datatype and object properties. The data is provided as supplementary material as gdc_cases.ttl [16], and contains case information about 33,549 patients that are currently available in the portal as of March, 2019. The RDF data was converted from the cases' API endpoint. The structure of the study information follows major classes and properties from SIO, as shown in Figure 1. We resolve data values to classes in NCI Thesaurus [6] using the BioPortal Annotator [13]. The resulting RDF is loaded into a triple store, along with the summary statistics, which is also represented in RDF.

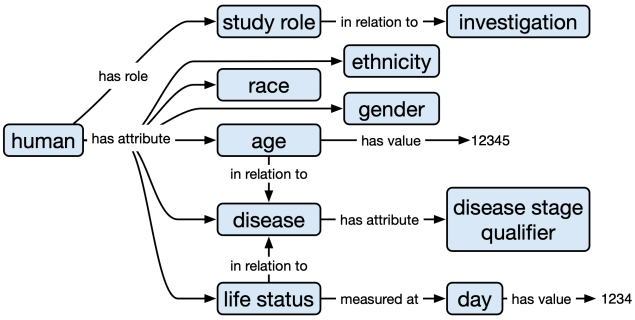

Fig. 1. A conceptual map of the GDC patient case information. We incorporate demographics such as race, ethnicity, gender, age at diagnosis, life (vital) status, disease type, and survival duration, and a link to the specific GDC investigation. This re-uses predicates and classes from SIO and is extended using classes from the NCI Thesaurus, to which the GDC data has been aligned.

3 Grouping Criteria as Classes

In order to create aggregate function semantics, we first have to formally define the grouping criteria. Fortunately, the grouping criteria used in OWL restrictions covers most cases. We therefore define an aggregate value as an attribute

of an OWL Class. The aggregation semantics from Calvanese *et al.* [2] effectively introduce variables into OWL class definitions. A conventional OWL class contains references to classes, properties, individuals, and literals:[5]

```
Class: GDC_Subject
EquivalentTo: sio:human
  and sio:'has role' some (sio:'subject role'
    and sio:'in relation to' some sio:investigation)
```

This can be expressed as this SPARQL query:

```
select ?GDC_Subject WHERE {
  ?GDC_Subject a sio:SIO_000485; # human
   sio:SIO_000228 [ # has role
      a sio:SIO_000883; # study subject
      sio:SIO_000668 [ # in relation to
        a sio:SIO_000747 # investigation
      ]
    ].
}
```

When this class is applied to the sample data in Supplemental Materials it results in 33,549 matches, one for each human:

$$GDC\_Subject \supseteq \left\{ \begin{array}{c} \text{case:d4f90900-3b81-4015-8e11-4b4525345063} \\ \text{case:d52a195d-7d63-4eb6-81c2-3c473ba57979} \\ \dots \end{array} \right\}$$

An aggregate query in Calvanese *et al.* is expressed as:

$$q\left(\bar{x}, \alpha\left(\bar{y}\right)\right) \leftarrow \phi$$

where $\bar{x}$ is a sequence of grouping variables, $\alpha\left(\bar{y}\right)$ is the aggregation term, and ϕ is the query condition expressed in first order logic. We translate this into manchester notation through the following template:

`Class:` $\bar{x}$
`SubClassOf:` ϕ

[5] The following prefixes are used in all SPARQL, Manchester notation, and Turtle examples:
rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#
rdfs: http://www.w3.org/2000/01/rdf-schema#
sio: http://semanticscience.org/resource/
prov: http://www.w3.org/ns/prov#
case: http://example.com/gdc/case/
project: http://example.com/gdc/project/

We will introduce $\bar{y}$ in Section 5, as Calvanese *et al.* do not provide a way to represent $\bar{y}$ in a knowledge graph. In order to more explicitly treat their uses of variables, we define a function $G(g_1, \ldots, g_n) = \bar{x}$:

```
Class: G( g1, ..., gn)
SubClassOf: φ
```

An aggregate query of study subjects by investigation can now be expressed as:

```
Class: G(?x)
SubClassOf: sio:human
  and sio:'has role' some (sio:'subject role'
    and sio:'in relation to' value ?x)
```

The selection SPARQL query would look like this:

```
select ?GDC_Subject ?x WHERE {
  ?GDC_Subject a sio:SIO_000485; # human
   sio:SIO_000228 [ # has role
      a sio:SIO_000883; # study subject
      sio:SIO_000668 ?x # in relation to
    ].
  ?x a sio:SIO_000747 # investigation
}
```

A class is defined for every matched value in the knowledge base:

```
Class: G(case:FM-AD)
EquivalentTo: sio:human
  and sio:'has role' some (sio:'subject role'
    and sio:'in relation to' value case:FM-AD)

Class: G(case:TARGET-NBL)
EquivalentTo: sio:human
  and sio:'has role' some (sio:'subject role'
    and sio:'in relation to' value case:TARGET-NBL)

...
```

The members of each grouping criterion $G(g_1, \ldots, g_n)$ are therefore members of the generated class $G()$, and the RDF graph that describes these summary statistics can provide *rdf:type* statements for each member as provenance. There are 45 different investigations in GDC total, above we show the three with the most subjects. These variables can replace classes, properties, and individuals, and can be mixed in with non-variable criteria, as was shown in the above example. Calvanese *et al.* discuss the computation of the aggregate operations MIN, MAX, COUNT DISTINCT, SUM, and AVG (mean), but do not specify how the values relate to the computed classes. The following sections relate the work done by Calvanese *et al.* to a complete representation of both the grouping

criteria and aggregate statistics in RDF. This includes methods for computable URIs for the grouping criteria classes $G()$, how to provide aggregate statistics on all $G()$ using *sio:'has attribute'*, and how to link those aggregate statistics to their source graphs.

4 Computable URIs for Grouping Criteria

We use the following method for computing URIs for G, which allows for alignment of grouping criteria independent of the source graph or individual naming schemes. Take the concise bounded description (CBD, or $C()$), minus annotations, of G in a separate RDF graph $C(G)$, where G itself is represented as a blank node. A CBD is the direct connections of a resource in the graph along with the transitive direct connections of any blank nodes in the description. Compute the graph digest of $C(G)$ and use the digest value to rewrite the URI for G in the original graph. The CBD means that the URI should be computable in any case. The URI will be different if the inferred closure of statements on G is computed instead of the minimal grouping critieria, so users will need to maintain consistency there.

5 Generating Facts About Classes using Aggregate Attributes

We use OWL 2 punning [23, 10] to provide natural metamodeling of aggregate attributes of classes and to reify non-SIO-based triples into a SIO-compatible format. "Punning" here refers to the ability to separate OWL Classes, Properties, and Individuals, even if they share the same URI. For instance, giving OWL class a meta-type in addition to *owl:Class* would place that ontology in OWL-Full, as would any facts (as opposed to annotations).

Once a grouping operation has been performed over a set of data to produce *owl:Classes*, it is now possible to compute aggregate functions over the members of the class. The aggregate functions available in SIO include mean, median, standard deviation, count, mode, minimal and maximal values, and can be extended with new statistical concepts using similar representation patterns. This is accomplished by using the following steps, with predicates mapped to OWL properties as shown in Table 1:

1. Define an OWL class that is a subclass of the aggregation criteria, as shown in Section 3.
2. Pun the *owl:Class* to an *owl:Individual* in order to make assertions about it. In RDF nothing needs to be explicitly done to do this.

3. If the predicate is not a subproperty of *sio:'is related to'*, reify the predicate of the values being aggregated using the following rules:

$$\forall S, P, O \begin{pmatrix} P(S,O) \wedge P \not\subset rel \wedge \\ P \in \text{DatatypeProperty} \end{pmatrix} \Rightarrow \exists A \left(A \in P \wedge attr(S,A) \wedge val(A,O) \right)$$

$$\forall S, P, O \begin{pmatrix} P(S,O) \wedge P \not\subset rel \wedge \\ P \in \text{ObjectProperty} \end{pmatrix} \Rightarrow \exists A \left(A \in P \wedge role(S,A) \wedge to(A,O) \right) \tag{1}$$

Note that this operation is simply mapping a non-SIO property into the SIO framework. Punning is not needed for native SIO attributes.

We can now determine a way to represent the aggregation term $\alpha(\bar{y})$. This is expressed in RDF where $\bar{y}$ is an attribute of $G(g_1, \ldots, g_n)$, and each $\alpha(\bar{y})$ is the attribute of type α:

$$\forall G, \alpha(\bar{y}) \exists A \in \alpha, Y \in \bar{y} attr(G,Y) \wedge attr(Y,A) \wedge val(A, \alpha(\bar{y}))$$

Following the data in the Supplemental Materials, when *G(case:TCGA-BRCA)* is defined as an aggregate as expressed above:

```
Class: G(case:TCGA-BRCA)
SubClassOf: sio:human
  and sio:'has role' some (sio:'subject role'
    and sio:'in relation to' value case:TCGA-BRCA)
```

the aggregate facts can then be asserted about *G(case:TCGA-BRCA)* in this way:

```
G(case:TCGA-BRCA) sio:has-attribute
 [ a sio:count; sio:'has value' 1098 ],
 [ a sio:age;
   sio:'has attribute'
     [ a sio:mean; sio:'has value' 21582 ],
     [ a sio:'maximal value'; sio:'has value' 2009 ];
     [ a sio:'minimal value'; sio:'has value' 32872 ],
 ].
```

By providing formal representations for aggregations, it becomes possible to formally define them using grouping criteria. Additional facts can be provided about each set through aggregate functions, which can be extended with more sophisticated statistical functions. Since each aggregation becomes a defined and denoted thing, it becomes possible to provide the provenance of those definitions, which would include the members of the class and aggregate query used to define it. These classes and facts about these classes can now be defined automatically using grouping functions and instances. Because of this, OLAP-like tools can use and generate assertions about the aggregate sets that they produce through user interaction, since OLAP relies on GROUP BY, filtering criteria, and aggregation functions. These classes can then be subjected to statistical analysis

Predicate	Property	Definition
rel()	*sio:'is related to'*	A is related to B iff there is some relation between A and B..
attr()	*sio:'has attribute'*	A relation that associates an entity with an attribute where an attribute is an intrinsic characteristic such as a quality, capability, disposition, function, or is an externally derived attribute determined from some descriptor
val()	*sio:'has value'*	A relation between an informational entity and its actual value (numeric, date, text, etc).
role()	*sio:'has role'*	A relation between an entity and a role that it bears.
to()	*sio:'in relation to'*	A comparative relation to indicate that the instance of the class holding the relation exists in relation to another entity.
der()	*prov:wasDerivedFrom*	A derivation is a transformation of an entity into another, an update of an entity resulting in a new one, or the construction of a new entity based on a pre-existing entity.

Table 1. First Order Logic predicates mapped to SIO and PROV properties, with definitions. These predicates may be mapped to other equivalent properties as needed.

and the definitions can be re-applied to further datasets for hypothesis testing or published as nanopublications.

6 Closing Graphs Over Aggregates

Aggregation techniques face a number of challenges when dealing with the open world assumption. First, aggregation functions assume that the available data is complete and whole. For instance, asserting that a class has ten instances is implicitly means that ten known instances have been counted, but that number could be higher (because of unknown instances). Additionally, because of the non-unique naming assumption, if those instances are actually identical but not known to be, then the class may actually have fewer than ten instances. Expressing aggregate values is useful, but the open world assumption prevents any final conclusions about aggregate statistics, because there can always be more facts to be discovered. We need to close the graph to additional statements. A number of approaches have been proposed to compute the content digest of an RDF graph [20, 3, 17, 15], and an implementation of [17] has been published as part of RDFlib [18]. The algorithm in [17] has the additional benefit of efficiently computing reproducible identifiers for blank nodes within the graph, producing stable identifiers for all RDF graphs. The approach in [15] uses nanopublications to provide a mechanism for referencing RDF graph content by URI similar to the approach in [17], but its approach to blank node skolemization (by providing

a UUID for each blank node) means that the graph identifiers are not consistent. By creating different graph identifiers for the same graph, it becomes impossible to verify that identical graphs from different sources are potentially the same. By encoding the graph digest as part of a URI scheme, the aggregate attributes can be encoded with a *prov:wasDerivedFrom* link to the digest-identified graph:

$$\forall G, \alpha(\bar{y}), N \exists A \in \alpha, Y \in \bar{y} \left(attr\left(G, Y\right) \wedge attr\left(Y, A\right) \wedge val(A, \alpha(\bar{y})) \wedge der(A, N) \right)$$

This allows for the reuse of the classes G across aggregations, while providing evidence for computation of A, Y in specific graphs N. The grouping criteria on G can be re-applied to other datasets for further validation. For instance, a new graph can either be merged with the original, or computed separately for independent validation. Members of each $G()$ can be classified using OWL inference (due to the equivalence restrictions), and aggregation across those sets can be computed within the new members. Each aggregate value would, because of its derivation statement, be traceable to a specific RDF graph, and changes in the input can be validated simply by comparing URIs of the derivations to determine the reproducibility and/or repeatiblility of the aggregate value.

7 Related Work

Aggregation semantics in logic programming is well researched, [22, 21, 1] but in many ways, these works do not address how to integrate the *output* of aggregate functions into a knowledge representation that integrates with the original data. The HiFun query language provides the means to express analytic queries directly through a relational algebra, but does not provide a formalized knowledge representation in the process. The RDF Data Cube Vocabulary, or QB, expresses entities-in-context as information artifacts, instead of as attributes of entitites [5]. It has been used in a number of cases to create specialized representations of summary statistics [14, 19, 9]. The formalisms of aggregation aren't clear with this approach, as the aggregate values are not expressed in RDF. Additionally, RDF Data Cube Vocabulary treats statistical data as information artifacts, and does not support for realist representations of scientific data and knowledge [8].

8 Evaluation: Exploring the Genomic Data Commons

In most publications, aggregate statistics are expressed in human-readable tables or in figures. While human readers can conceptually make the connection back to the input data that has been aggregated, since no internal representation exists, the connection between source and aggregate are not maintained and cannot be used. Using our knowledge representation, we can query and persist those statistics as linked data that integrates with the original data. Using the supplementary materials script create_summaries.ipynb, we were able to compute

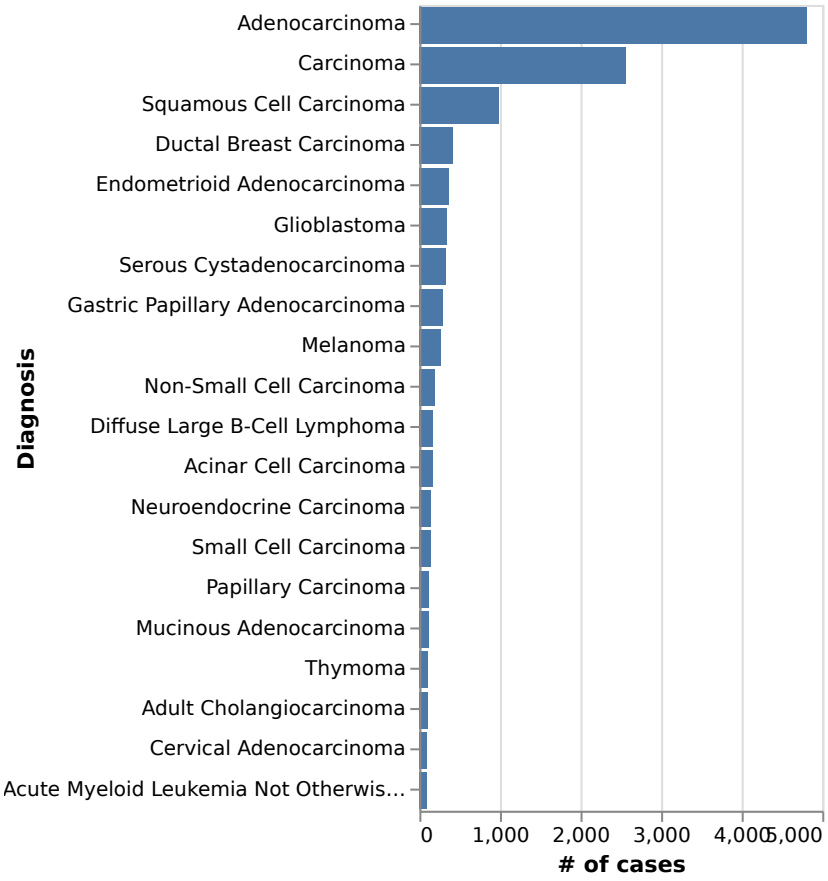

Fig. 2. The top 20 diagnoses for cancer in the GDC, retrieved from summary semantics. This figure was generated from the aggregate statistics encoded in age_by_diagnosis.ttl, and is reusable for viewing similar aggregations.

count statistics and age mean, min, and max by diagnosis[6]. The source data is contained in gdc_cases.nt. The class URIs for these aggregates are recomputable from their definitions, and we are able to construct summary visualizations using the graphs. Figure 2 contains summary counts for the top 20 diagnoses in GDC, retrieved from a summary semantics graph. This figure was generated from the RDF graph in the supplementary materials file age_by_diagnosis.ttl.

8.1 Description Logic Complexity

The Description Logic (DL) complexity of $G(g_1, \ldots, g_n)$ is not impacted by the way we express these classes, since they have no direct semantics in DL,

[6] age_by_diagnosis.ttl

and the DL-based definitions of each class are used to compute a URI for each $G()$. The generated $G(g_1, \ldots, g_n)$ classes are themselves in $\mathcal{ALE}$. SIO is in $\mathcal{SRIQ}(\mathcal{D})$. Additionally, were able to use Pellet to perform a full inference of the age_by_diagnosis.ttl with SIO without any inconsistencies or errors.

8.2 Overhead

The storage overhead of our knowledge representation is fairly limited. For instance, in our GDC dataset, expressing age by diagnosis for the entire dataset required 4,992 statements using 304 classes, requiring about 16 statements (with RDFS labels mixed in) per class. When using multiple grouping criteria, the number of expressed classes will expand geometrically based on the number of combined class criteria. The underlying data provided contained 8,048,537 statements, resulting in a significant reduction in statements.

9 Conclusions

The approach of OWL representations for grouping criteria plus SIO-based attributes for the summary statistics is a natural extension of both representations, and makes interoperability within Linked Data much easier. Assertions of aggregate information about groups of entities can now be formally expressed in ways that are traceable to their source. In some cases, these assertions can even be computed based on the available summary statistics of the data at hand. This improves the ability for researchers to build facts and hypotheses about their entities of interest from their data. This method builds on existing ontologies in provenance, such as Prov-O, and eScience, (e.g. SIO) and results in assertions with justifiable explanations. These assertions and their explanations can be published as nanopublications. The proposed knowledge representation is extensible across any aggregate functions that can be applied to defined sets of entities. This paper also provides a way of computing URIs for classes based on their necessary and sufficient conditions using graph digests of those OWL restrictions, making access of those parameterized classes stable across datasets. Finally, use of these aggregate semantics enables the use of existing analytical tools to generate explainable and exportable assertions about the data being analyzed and produce grouping criteria that can be re-applied to other datasets for further validation.

10 Acknowledgements

Thank you to James Michaelis and John Erickson for feedback and examples. This work is supported by IBM Research AI through the AI Horizons Network.

References

1. Afrati, F., Kolaitis, P.G.: Answering aggregate queries in data exchange. In: Proceedings of the twenty-seventh ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems. pp. 129–138. ACM (2008)
2. Calvanese, D., Kharlamov, E.: Aggregate Queries Over Ontologies. Proceedings of the 2nd international workshop on Ontologies and information systems for the semantic web (2008), `http://dl.acm.org/citation.cfm?id=1458484.1458500`
3. Carroll, J.J., Bizer, C., Hayes, P., Stickler, P.: Named graphs, provenance and trust. In: Proceedings of the 14th international conference on World Wide Web. pp. 613–622. ACM (2005)
4. Codd, E., Codd, S., Salley, C.: Providing OLAP (on-line analytical processing). Codd and Date (1993)
5. Cyganiak, R., Reynolds, D.: The RDF data cube vocabulary. W3C recommendation, W3C (Jan 2014), http://www.w3.org/TR/2014/REC-vocab-data-cube-20140116/
6. De Coronado, S., Haber, M.W., Sioutos, N., Tuttle, M.S., Wright, L.W., et al.: NCI thesaurus: using science-based terminology to integrate cancer research results. In: Medinfo. pp. 33–37 (2004)
7. Dumontier, M., Baker, C.J., Baran, J., Callahan, A., Chepelev, L., Cruz-Toledo, J., Del Rio, N.R., Duck, G., Furlong, L.I., Keath, N., et al.: The semanticscience integrated ontology (sio) for biomedical research and knowledge discovery. Journal of biomedical semantics **5**(1), 14 (2014)
8. Dumontier, M., Hoehndorf, R.: Realism for scientific ontologies. In: FOIS. pp. 387–399 (2010)
9. Ermilov, I., Martin, M., Lehmann, J., Auer, S.: Linked open data statistics: Collection and exploitation. In: International Conference on Knowledge Engineering and the Semantic Web. pp. 242–249. Springer (2013)
10. Grau, B.C., Horrocks, I., Motik, B., Parsia, B., Patel-Schneider, P., Sattler, U.: Owl 2: The next step for owl. Web Semantics: Science, Services and Agents on the World Wide Web **6**(4), 309–322 (2008)
11. Grossman, R.L., Heath, A.P., Ferretti, V., Varmus, H.E., Lowy, D.R., Kibbe, W.A., Staudt, L.M.: Toward a shared vision for cancer genomic data. New England Journal of Medicine **375**(12), 1109–1112 (2016)
12. Ihaka, R., Gentleman, R.: R: A language for data analysis and graphics. Journal of computational and graphical statistics **5**(3), 299–314 (1996)
13. Jonquet, C., Shah, N.H., Musen, M.A.: The open biomedical annotator. Summit on translational bioinformatics **2009**, 56 (2009)
14. Kämpgen, B., ORiain, S., Harth, A.: Interacting with statistical linked data via olap operations. In: Extended Semantic Web Conference. pp. 87–101. Springer (2012)
15. Kuhn, T., Dumontier, M.: Trusty uris: Verifiable, immutable, and permanent digital artifacts for linked data. In: European semantic web conference. pp. 395–410. Springer (2014)
16. McCusker, J.: Supplementary Materials for A Linked Data Representation for Summary Statistics and Grouping Criteria (2019). https://doi.org/10.7910/DVN/OK0BUG
17. McCusker, J.P.: WebSig: a digital signature framework for the web. Ph.D. thesis, Rensselaer Polytechnic Institute (2015)

18. RDFLib Team: rdflib 4.2.2 (2013), `https://rdflib.readthedocs.io`, accessed 4/1/2019
19. Salas, P.E.R., Martin, M., Da Mota, F.M., Auer, S., Breitman, K., Casanova, M.A.: Publishing statistical data on the web. In: 2012 IEEE Sixth International Conference on Semantic Computing. pp. 285–292. IEEE (2012)
20. Sayers, C., Karp, A.H.: Computing the digest of an rdf graph. Mobile and Media Systems Laboratory, HP Laboratories, Palo Alto, USA, Tech. Rep. HPL-2003-235 **1** (2004)
21. Sudarshan, S., Srivastava, D., Ramakrishnan, R., Beeri, C.: Extending the well-founded and valid semantics for aggregation. In: ILPS. pp. 590–608 (1993)
22. Van Gelder, A.: The well-founded semantics of aggregation. In: Proceedings of the eleventh ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems. pp. 127–138. ACM (1992)
23. Wallace, E., Golbreich, C.: OWL 2 Web Ontology Language New Features and Rationale (Second Edition). W3C recommendation, W3C (Dec 2012), http://www.w3.org/TR/2012/REC-owl2-new-features-20121211/