Balder Cancer Type Matching - Analysis Report

Overview

This document contains analysis reporting for the Balder 'cancer_type_matching' procedure. The goal of this work is to apply a cross-ontology framework to improve mapping of cancer types represented in both genomic data sets and clinically actionability biomarker databases.

Loading data

This notebook works by first loading the data generated in "cancer_type_matching.r' into memory

```
inRFile <- "/Users/larsonhogstrom/Documents/oncology_biomarkers/cancer_type_matching/cance
load(inRFile)
```

Methods overview

1. Load MySQL data products for genomic datasets and actionability databases
2. Create a table of "source cancer types" which contains all unique cancer type strings from various each source from step 1)
3. Create a lower case string and remove substrings (such as "cancer" or "disease") to prepare for query
4. Load PhenOncoX records database of cancer type terms
5. Get a representative phenoOncoX ID for each source term
    1. Loop through each unique source term

    2. For each unique term, create a lower case string and remove substrings (such as “cancer” or “disease”) to prepare for query
    3. Identify string matches in the phenOncoX record table. There are multiple columns of terms in the record table. Create a boolean matrix of match/no-match for each source term
    4. Join the record information to the matrix of matches
    5. select a representative phenOncoX record for each unique source term: max hit or max level
    6. identify which column drove the hit and make summary plots
6. Make pairwise comparisons between each of the databases.
    1. How many original source terms from db X are found in db Y
    2. How many phenOncoX records from db X are found in db Y
7. create heatmap showing pairwise comparisons across all DBs

Source term search approach

Eight search types were tested.

Definition 1: string modification

Variables defining if source term strings are modified:

- **modifiedSourceString** = For each unique term, create a lower case string and remove substrings (such as “cancer” or “disease”) to prepare for query.
- **origSourceString** = unmodified cancer type string from source

Definition 2: Exact or substring match

Approach for text matching to any given phenOncoX record:

- **substringMatch** = Source term is allowed to be a substring of the record term when testing for a match.
- **exactStringMatch** = source term must match record entry exactly

Definition 3: representative entry selection

For each unique source term, pick a single representative PhenoOncoX entry match based on either:

- **MaxHits** = PhenoOncoX entry where the source term was contained in the largest number of fields
- **MaxLevel** = a PhenoOncoX entry that matched on at least one column and had the largest oncotree level

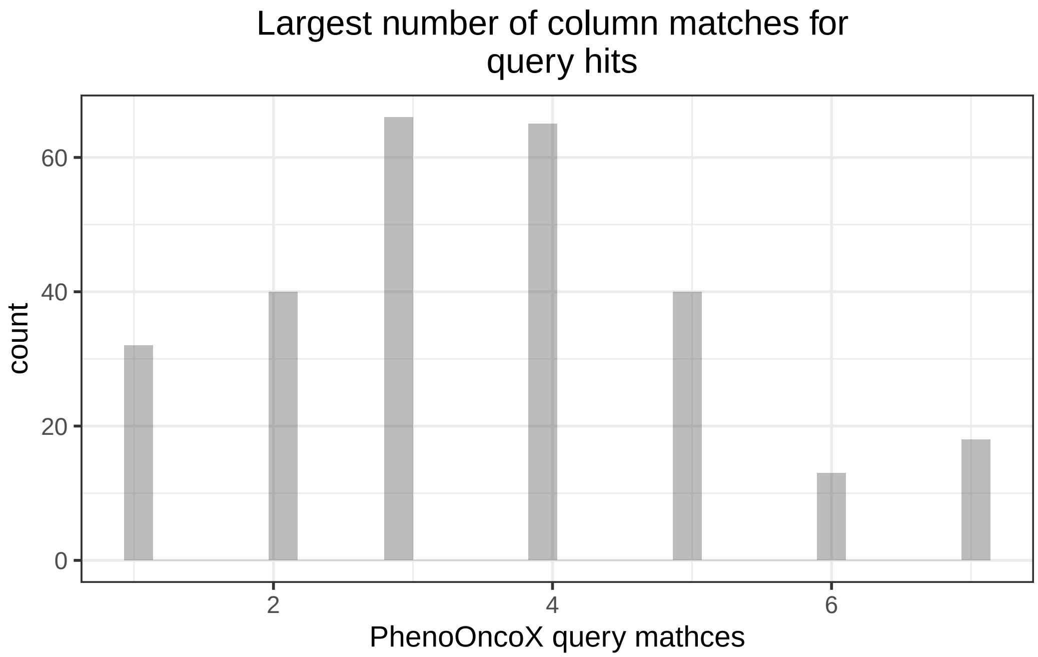
The following table shows entry counts after source search in phenOncoX record matching:

		modifiedSourceString	origSourceString
exactStringMatch	MaxHit	149 (50%)	98 (32%)
exactStringMatch	MaxLevel	149 (50%)	98 (32%)
substringMatch	MaxHit	274 (92%)	216 (72%)
substringMatch	MaxLevel	274 (92%)	216 (72%)

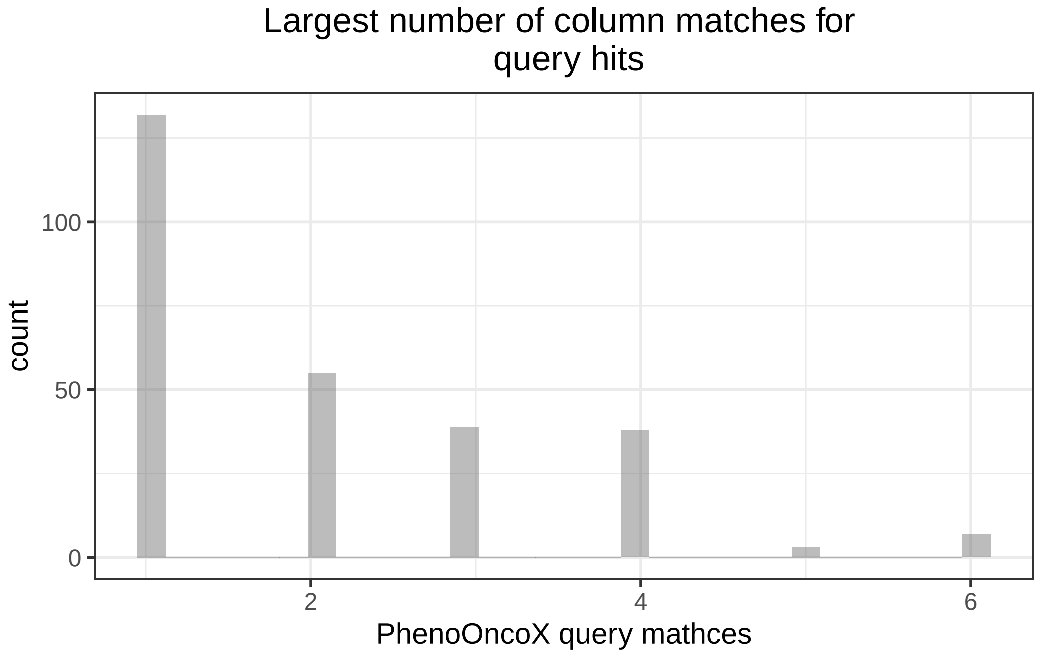
For the rest of the document, results for the following approach are highlighted: SubstringMatch & MaxLevel & .modifiedSourceString

Results summary

Plot the number of query hits, plot the oncotree higherarchy number



Max level results



Pairwise comparisons of cancer type source terms

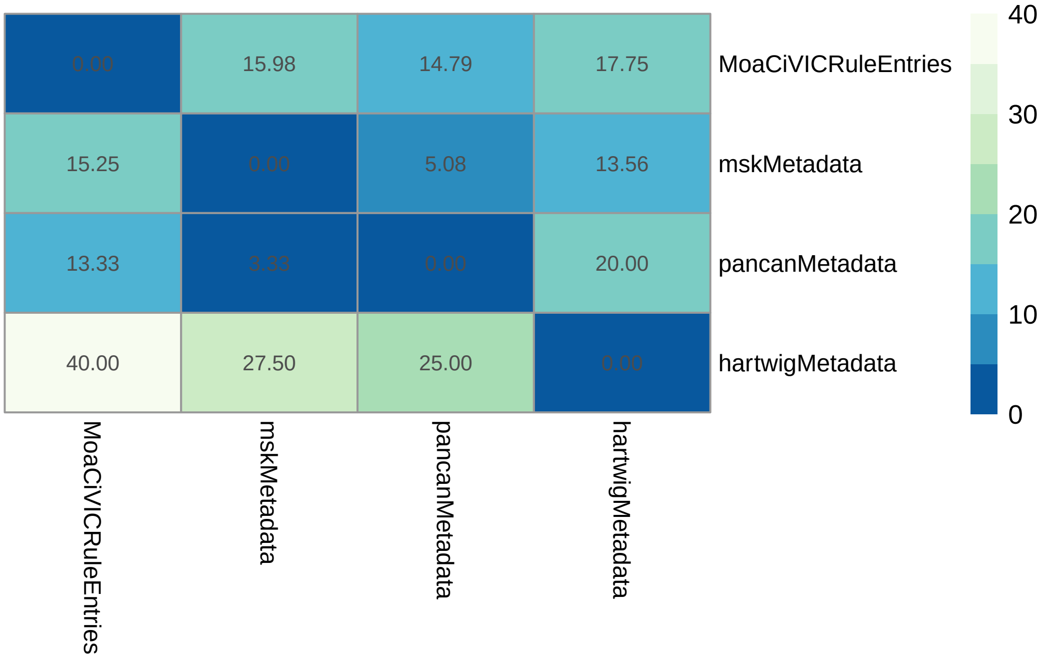


Table for MSK to MOA/CIViC counts by cancer type matching approach

Computing actionable variant counts for different match approaches

Three different actionablity matching approaches were computed:

1. Unrestricted where no cancer type matching was imposed
2. Restricted with raw cancer type string matching
3. Restricted with ontology-based cancer type matching

Reporting table

Cancer type matching approach	Cancer type match restriction	number of variants	percent of actionable mutations	number of patients	percent of patients
Raw string match	Restricted	69	17.1%	1789	18.6%

Cancer type matching approach	Cancer type match restriction	number of variants	percent of actionable mutations	number of patients	percent of patients
Ontology-based match	Restricted	78	19.4%	1950	20.3%
N/A	Unestricted	117	29%	4437	46.3%

Why are ontology-based matching rates lower for restricted actionability events?

- Are there any patients with actionable mutations found in raw matching but not ontology-based?

```
FALSE
62732
```

```
iOntology.based.unique.patients
FALSE  TRUE
64256   655
```

Where was the benefit seen for the ontology-based approach? What new types of matches did it enable?

The were 655 unique actionability entries found with the ontology-based approach, but not with raw cancer type matching. These matches derived from 160 unique subjects.

The following table shows the breakdown of patient count improvements:

cancer type	number of patients
Thyroid Cancer	81
Non-Small Cell Lung Cancer	58
Mature B-Cell Neoplasms	8
Histiocytosis	5
Hepatobiliary Cancer	4
Glioma	3
Bladder Cancer	1
Small Cell Lung Cancer	1

Were these cancer types in the raw text?

```
`summarise()` has grouped output by 'CANCER_TYPE.x', 'CANCER_TYPE.y'. You can
override using the `.groups` argument.
```

cancer type - MSK	cancer type - CIVIC/MOA	phenOncoX record ID	number of patients
Thyroid Cancer	Thyroid Gland Cancer	22740	81
Non-Small Cell Lung Cancer	Lung Cancer	10315	58
Mature B-Cell Neoplasms	Follicular Lymphoma	13494	8
Histiocytosis	Erdheim-Chester Disease	14884	5
Hepatobiliary Cancer	Cancer	415	4
Glioma	Glioblastoma	3648	3
Bladder Cancer	Cancer	415	1
Small Cell Lung Cancer	Lung Cancer	10315	1

List out corresponding phenoOncoX records

22740

What proportion of entries were cancer type entries were in MSK?