Personalize Oncology

In [2]:

```
from IPython.display import IFrame
IFrame(src="https://terra.bio/the-molecular-oncology-almanac-from-algorithm-to-analysis-portal/", width='100%', height='500px')
```

Out[2]:

(https://terra.bio) MENU

The Molecular Oncology

Certain type of Cancer was suggested (Melanoma). We will leverage Molecular Oncology Almanac Search platform from Verely: https://terra.bio/the-molecular-oncology-almanac-from-algorithm-to-analysis-portal/ (https://terra.bio/the-molecular-oncology-almanac-from-algorithm-to-analysis-portal/) to explore potential available therapies suggested for different mutations

In [3]:

```
from IPython.display import IFrame
IFrame(src="https://moalmanac.org/search?s=%22Melanoma%22%5Bdisease%5D", width='100%', hei
ght='400px')
```

Out[3]:

Molecular Oncology Almanac Search Results

Search

Multiple search terms may be combined. Click here for search help.

Click on any alteration below to view more details about the alteration-actionability relationship.

Feature type	Feature	Therapy
Somatic Variant (search? s="Somatic%20Variant" [feature])	BRAF (/search? s=Gene%3A%22BRAF%22%5Battribute%5D) p.V600E (Missense)	Dabrafeni [therapy])

Explore personal Genome for mutations

- Use GDC Data portal - [https://portal.gdc.cancer.gov/ (https://portal.gdc.cancer.gov/)](https://portal.gdc.cancer.gov/)
  - Explore all specific mutations whoich might be present in patient genome - [https://portal.gdc.cancer.gov/ssms/92469838-fc8e-5a01-af9f-10268c739bb3 (https://portal.gdc.cancer.gov/ssms/92469838-fc8e-5a01-af9f-10268c739bb3)](https://portal.gdc.cancer.gov/ssms/92469838-fc8e-5a01-af9f-10268c739bb3)
    - Find Rs number in dbSNP - [https://www.ncbi.nlm.nih.gov/snp/rs121913513 (https://www.ncbi.nlm.nih.gov/snp/rs121913513)](https://www.ncbi.nlm.nih.gov/snp/rs121913513)
    - find mutation details in Snpedia - [https://www.snpedia.com/index.php/Rs121913513 (https://www.snpedia.com/index.php/Rs121913513)](https://www.snpedia.com/index.php/Rs121913513)

In [15]:

```
from IPython.display import IFrame
IFrame(src="https://www.snpedia.com/index.php/Rs121913513", width='100%', height='400px')
```

Out[15]:

SNPedia

rs121913513

Orientation	plus	
Stabilized	plus	
Geno	**Mag**	**Summary**
(T;T)	0	common in clinvar
Make rs121913513(C;C)		
Make rs121913513(C;T)		
Reference	GRCh38.p7 38.3/150	
Chromosome	4	
Position	54727495	
Gene	KIT	

Search in patient genome for mutations

In [9]:

```
!head -50 AncestryDNAtest.txt
```

```
#AncestryDNA raw data download
#This file was generated by AncestryDNA at: 10/20/2020 22:03:06 UTC
#Data was collected using AncestryDNA array version: V2.0
#Data is formatted using AncestryDNA converter version: V1.0
#Below is a text version of your DNA file from Ancestry.com DNA, LLC.  THIS
#INFORMATION IS FOR YOUR PERSONAL USE AND IS INTENDED FOR GENEALOGICAL RESEAR
CH
#ONLY.  IT IS NOT INTENDED FOR MEDICAL, DIAGNOSTIC, OR HEALTH PURPOSES.  THE
EXPORTED DATA IS
#SUBJECT TO THE ANCESTRY TERMS AND CONDITIONS, BUT PLEASE BE AWARE THAT THE
#DOWNLOADED DATA WILL NO LONGER BE PROTECTED BY OUR SECURITY MEASURES.
#WHEN YOU DOWNLOAD YOUR RAW DNA DATA, YOU ASSUME ALL RISK OF STORING,
#SECURING AND PROTECTING YOUR DATA.  FOR MORE INFORMATION, SEE ANCESTRYDNA FA
QS.
#
#Genetic data is provided below as five TAB delimited columns.  Each line
#corresponds to a SNP.  Column one provides the SNP identifier (rsID where
#possible).  Columns two and three contain the chromosome and basepair positi
on
#of the SNP using human reference build 37.1 coordinates.  Columns four and f
ive
#contain the two alleles observed at this SNP (genotype).  The genotype is re
ported
#on the forward (+) strand with respect to the human reference.
rsid    chromosome      position        allele1 allele2
rs3131972       1       752721  G       G
rs114525117     1       759036  G       G
rs4040617       1       779322  A       A
rs141175086     1       780397  C       C
rs115093905     1       787173  G       G
rs11240777      1       798959  G       G
rs6681049       1       800007  C       C
rs4422948       1       835499  A       A
rs57494724      1       837192  A       A
rs4475691       1       846808  C       C
rs6657440       1       850780  T       C
rs4970461       1       852964  T       G
rs7537756       1       854250  A       A
rs13302982      1       861808  G       G
rs2880024       1       866893  C       C
rs74047407      1       866938  G       G
rs1110052       1       873558  T       T
rs7523549       1       879317  C       C
rs2272756       1       882033  G       G
rs3748597       1       888659  C       C
rs13302957      1       891021  A       A
rs13303106      1       891945  G       G
rs13303010      1       894573  A       A
rs200468119     1       899489  G       G
rs6696281       1       903104  C       C
rs6696609       1       903426  C       C
rs28391282      1       904165  G       G
rs28477686      1       910394  C       C
rs6660139       1       916549  G       G
rs13303118      1       918384  T       T
rs2341354       1       918573  G       G
```

In [4]:

```
!grep 'rs121913513' AncestryDNAtest.txt
```

```
rs121913513     4       55593661        T       T
```

If mutation found for a patient than oncologist might want to explore proposed therapy

In [ ]:

```

```

Non-oncology drugs pepurposing

In [ ]:

```

```

Drug Repurposing via disease-compounds relations

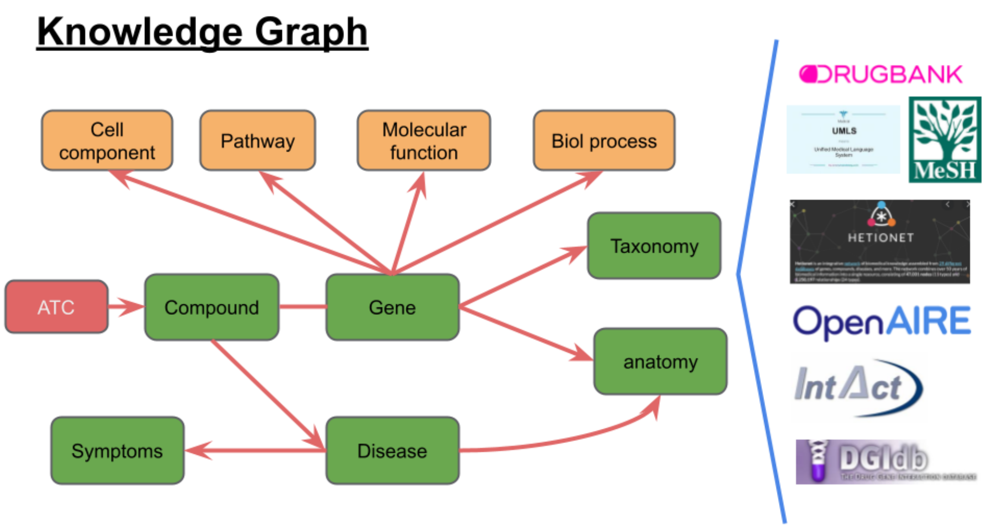
This example shows how to do drug repurposing using DRKG even with the pretrained model.

This can be used to detect Medical claim fraud when claim does not pass the check "symptopms-disease-prescribed medication"

In [12]:

```
from IPython.display import Image
Image(filename='./kg.png', width = 500, height = 300)
```

Out[12]:



In [ ]:

```

```

Collecting related disease

At the very beginning we need to collect a list of disease in DRKG. We can easily use the Disease ID that DRKG uses for encoding the disease. References:

```
https://meshb.nlm.nih.gov/search?searchInField=allTerms&sort=&size=20&searchType=anyWord&searchMethod=FullWord&q=Macular%20Degeneration

https://go.drugbank.com/drugs/DB00536

https://en.wikipedia.org/wiki/Aging-associated_diseases

https://www.verywellhealth.com/age-related-diseases-2223996
```

Here we take all of the age-related disease as target:

```
1. Aging - D000375 -index not excepted

2. Age-Related Macular Degeneration (AMD) - D008268
    SNP-
            https://pubmed.ncbi.nlm.nih.gov/27099955/

            https://jamanetwork.com/journals/jamaophthalmology/fullarticle/2514061
            https://www.snpedia.com/index.php/Rs1061170
            https://www.snpedia.com/index.php/Rs10490924
                ....

3. Cataracts - D002386

4. Glaucoma - D005901

5. Chronic Obstructive Pulmonary Disease - D029424

6. Atherosclerosis - D003324

7. Cardiovascular disease - D002318

8. Stroke - D020521

9. Heart Diseases - D006331

10. Hypertension - D006973

11. Hypertension, Pulmonary - D006976

12. Myocardial Ischemia - D017202

13. Brain Ischemia - D002545

14. Cancer (Neoplasms) - D009369

15. Colorectal Neoplasms - D003123

16. Arthritis - D001172

17. Spondylarthritis - D025241

18. Osteoarthritis - D010003

19. Osteoporosis - D010024

20. Diabetes Mellitus - D003920

21. Diabetes Mellitus, Type 2 - D003924
```

In [56]:

```
Aging_disease_list = [
'Disease::MESH:D008268',
'Disease::MESH:D002386',
'Disease::MESH:D005901',
'Disease::MESH:D029424',
'Disease::MESH:D003324',
'Disease::MESH:D002318',
'Disease::MESH:D020521',
'Disease::MESH:D006331',
'Disease::MESH:D006973',
'Disease::MESH:D006976',
'Disease::MESH:D017202',
'Disease::MESH:D002545',
'Disease::MESH:D009369',
'Disease::MESH:D003123',
'Disease::MESH:D001172',
'Disease::MESH:D025241',
'Disease::MESH:D010003',
'Disease::MESH:D010024',
'Disease::MESH:D003920',
'Disease::MESH:D003924',
'Disease::MESH:D000544',
'Disease::MESH:D003704',
'Disease::MESH:D057180',
'Disease::MESH:D020961',
'Disease::MESH:D006816',
'Disease::MESH:D010300',
'Disease::MESH:D057174',
'Disease::MESH:D011470',
'Disease::MESH:D053201',
'Disease::MESH:D005705'
]
```

In [57]:

```
# Aging_disease_list = ['Disease::MESH:D010003']
```

In [58]:

```
# Aging_disease_list = ['Disease::MESH:D000544']
```

We are now will explore drug repositioning for Melonoma - https://meshb.nlm.nih.gov/record/ui?ui=D008545 (https://meshb.nlm.nih.gov/record/ui?ui=D008545)

In [59]:

```
Aging_disease_list = ['Disease::MESH:D008545']
```

Candidate drugs

Now we use FDA-approved drugs in Drugbank as candidate drugs. (we exclude drugs with molecule weight < 250) The drug list is in infer_drug.tsv

In [60]:

```
import csv

# Load entity file
drug_list = []
with open("./infer_drug.tsv", newline='', encoding='utf-8') as csvfile:
    reader = csv.DictReader(csvfile, delimiter='\t', fieldnames=['drug','ids'])
    for row_val in reader:
        drug_list.append(row_val['drug'])
```

In [61]:

```
len(drug_list)
```

Out[61]:

```
8104
```

Treatment relation

Two treatment relations in this context

In [62]:

```
treatment = ['Hetionet::CtD::Compound:Disease','GNBR::T::Compound:Disease']
```

Get pretrained model

We can directly use the pretrianed model to do drug repurposing.

In [63]:

```
import pandas as pd
import numpy as np
import sys
sys.path.insert(1, '../utils')
from utils import download_and_extract
download_and_extract()
```

In [64]:

```
entity_idmap_file = '../data/drkg/embed/entities.tsv'
relation_idmap_file = '../data/drkg/embed/relations.tsv'
```

In [65]:

```
entity_idmap_file = '../data/embed/entities.tsv'
relation_idmap_file = '../data/embed/relations.tsv'
```

Get embeddings for diseases and drugs

In [66]:

```
# Get drugname/disease name to entity ID mappings
entity_map = {}
entity_id_map = {}
relation_map = {}
with open(entity_idmap_file, newline='', encoding='utf-8') as csvfile:
    reader = csv.DictReader(csvfile, delimiter='\t', fieldnames=['name','id'])
    for row_val in reader:
        entity_map[row_val['name']] = int(row_val['id'])
        entity_id_map[int(row_val['id'])] = row_val['name']

with open(relation_idmap_file, newline='', encoding='utf-8') as csvfile:
    reader = csv.DictReader(csvfile, delimiter='\t', fieldnames=['name','id'])
    for row_val in reader:
        relation_map[row_val['name']] = int(row_val['id'])

# handle the ID mapping
drug_ids = []
disease_ids = []
for drug in drug_list:
    drug_ids.append(entity_map[drug])

for disease in Aging_disease_list:
    disease_ids.append(entity_map[disease])

treatment_rid = [relation_map[treat]  for treat in treatment]
```

In [67]:

```
# Load embeddings
import torch as th
entity_emb = np.load('../data/drkg/embed/DRKG_TransE_l2_entity.npy')
rel_emb = np.load('../data/drkg/embed/DRKG_TransE_l2_relation.npy')

drug_ids = th.tensor(drug_ids).long()
disease_ids = th.tensor(disease_ids).long()
treatment_rid = th.tensor(treatment_rid)

drug_emb = th.tensor(entity_emb[drug_ids])
treatment_embs = [th.tensor(rel_emb[rid]) for rid in treatment_rid]
```

Drug Repurposing Based on Edge Score

We use following algorithm to calculate the edge score. Note, here we use logsigmiod to make all scores < 0. The larger the score is, the stronger the h will have r with t.

$\mathbf{d} = \gamma - ||\mathbf{h}+\mathbf{r}-\mathbf{t}||_{2}$

$\mathbf{score} = \log\left(\frac{1}{1+\exp(\mathbf{-d})}\right)$

When doing drug repurposing, we only use the treatment related relations.

In [68]:

```
import torch.nn.functional as fn

gamma=12.0
def transE_l2(head, rel, tail):
    score = head + rel - tail
    return gamma - th.norm(score, p=2, dim=-1)

scores_per_disease = []
dids = []
for rid in range(len(treatment_embs)):
    treatment_emb=treatment_embs[rid]
    for disease_id in disease_ids:
        disease_emb = entity_emb[disease_id]
        score = fn.logsigmoid(transE_l2(drug_emb, treatment_emb, disease_emb))
        scores_per_disease.append(score)
        dids.append(drug_ids)
scores = th.cat(scores_per_disease)
dids = th.cat(dids)
```

In [69]:

```
# sort scores in decending order
idx = th.flip(th.argsort(scores), dims=[0])
scores = scores[idx].numpy()
dids = dids[idx].numpy()
```

Now we output proposed treatments

In [70]:

```
_, unique_indices = np.unique(dids, return_index=True)
topk=100
topk_indices = np.sort(unique_indices)[:topk]
proposed_dids = dids[topk_indices]
proposed_scores = scores[topk_indices]
```

Now we list the pairs of in form of (drug, treat, disease, score)

We select top K relevent drugs according the edge score

In [71]:

```
for i in range(topk):
    drug = int(proposed_dids[i])
    score = proposed_scores[i]

    print("{}\t{}".format(entity_id_map[drug], score))
```

```
Compound::DB00515       -0.10456792265176773
Compound::DB09107       -0.11128311604261398
Compound::DB08894       -0.11279210448265076
Compound::DB00441       -0.11503596603870392
Compound::DB00997       -0.11634162813425064
Compound::DB11672       -0.11968539655208588
Compound::DB01229       -0.12061229348182678
Compound::DB00675       -0.12224452942609787
Compound::DB00877       -0.12437762320041656
Compound::DB11890       -0.12975651025772095
Compound::DB00755       -0.13055695593357086
Compound::DB06159       -0.1350637972354889
Compound::DB00563       -0.14131759107112885
Compound::DB04786       -0.14245563745498657
Compound::DB11678       -0.145339220762252B
Compound::DB00290       -0.14617787301540375
Compound::DB01234       -0.1529041975736618
Compound::DB01181       -0.15405192971229553
Compound::DB00958       -0.1550951600074768
Compound::DB04106       -0.15705730020999908
Compound::DB00570       -0.15721775591373444
Compound::DB00773       -0.15808223187923431
Compound::DB00305       -0.15916049480438232
Compound::DB00143       -0.16185718774795532
Compound::DB12983       -0.16363579034805298
Compound::DB13811       -0.16527362167835236
Compound::DB04690       -0.16603969037532806
Compound::DB00541       -0.167564511299133B
Compound::DB04849       -0.17062176764011383
Compound::DB06589       -0.17079845070838928
Compound::DB12028       -0.17334671318531036
Compound::DB12116       -0.173793762922287
Compound::DB14765       -0.17381730675697327
Compound::DB00641       -0.1739695817232132
Compound::DB12832       -0.17500224709510803
Compound::DB11996       -0.17527860403060913
Compound::DB05382       -0.1768036037683487
Compound::DB00993       -0.17840565741062164
Compound::DB01041       -0.17842061817646027
Compound::DB12916       -0.1793442815542221
Compound::DB04216       -0.18056678771972656
Compound::DB01645       -0.18087345361709595
Compound::DB00970       -0.1831292062997818
Compound::DB06287       -0.18473733961582184
Compound::DB00783       -0.18533086776733398
Compound::DB01204       -0.1872987002134323
Compound::DB01042       -0.18762345612049103
Compound::DB00650       -0.18975037336349487
Compound::DB04540       -0.19029483199119568
Compound::DB11094       -0.19053132832050323
Compound::DB11651       -0.1906197965145111
Compound::DB13647       -0.19371041655540466
Compound::DB00624       -0.19406747817993164
Compound::DB11797       -0.19738216698169708
Compound::DB03459       -0.19858995079994202
Compound::DB00091       -0.1996825635433197
Compound::DB05294       -0.20092160999774933
```

```
Compound::DB00328        -0.20302289724349976
Compound::DB05022        -0.20339734852313995
Compound::DB09341        -0.20512278378009796
Compound::DB00694        -0.20791608095169067
Compound::DB06641        -0.21561232209205627
Compound::DB01211        -0.21589046716690063
Compound::DB00501        -0.21655230224132538
Compound::DB01611        -0.2167699933052063
Compound::DB01030        -0.21689735352993011
Compound::DB00445        -0.21691367030143738
Compound::DB00158        -0.2196766585111618
Compound::DB00741        -0.2200285792350769
Compound::DB11759        -0.22050297260284424
Compound::DB00635        -0.22165098786354065
Compound::DB01045        -0.2221907526254654
Compound::DB03496        -0.22308099269866943
Compound::DB00982        -0.22325760126113892
Compound::DB01065        -0.22435645759105682
Compound::DB01394        -0.22484669089317322
Compound::DB01073        -0.22496169805526733
Compound::DB00132        -0.2265186309814453
Compound::DB09079        -0.22857406735420227
Compound::DB06486        -0.23413515090942383
Compound::DB09078        -0.23612013459205627
Compound::DB05134        -0.23617133498191833
Compound::DB00608        -0.23808418214321136
Compound::DB01177        -0.23896554112434387
Compound::DB00959        -0.23988288640975952
Compound::DB14674        -0.24032872915267944
Compound::DB00806        -0.24067749083042145
Compound::DB00291        -0.2428607940673828
Compound::DB01167        -0.2448461800813675
Compound::DB00396        -0.24519570171833038
Compound::DB00444        -0.2455003261566162
Compound::DB00688        -0.2455192655324936
Compound::DB11617        -0.24638701975345612
Compound::DB05273        -0.2466776818037033
Compound::DB02527        -0.2467825710773468
Compound::DB09073        -0.2477140873670578
Compound::DB06810        -0.2477719634771347
Compound::DB06263        -0.24821607768535614
Compound::DB01222        -0.24841062724590302
Compound::DB04297        -0.2491813749074936
```

Check Clinial Trial Drugs

There are several clinial trial drugs hit in top 100. (Note: Ribavirin exists in DRKG as a treatment for SARS). We need change this list to Kauffman protocol

In [72]:

```
clinical_drugs_file = './Disease_clinical_trial_drugs.tsv'
clinical_drug_map = {}
with open(clinical_drugs_file, newline='', encoding='utf-8') as csvfile:
    reader = csv.DictReader(csvfile, delimiter='\t', fieldnames=['id', 'drug_name','drug_i
d'])
    for row_val in reader:
        clinical_drug_map[row_val['drug_id']] = row_val['drug_name']

for i in range(topk):
    drug = entity_id_map[int(proposed_dids[i])][10:17]
    if clinical_drug_map.get(drug, None) is not None:
        score = proposed_scores[i]
        print("[{}]\t{}\t{}".format(i, clinical_drug_map[drug],score , proposed_scores[i
]))
```

```
[5]	Curcumin	-0.11968539655208588
[31]	EGCG	-0.173793762922287
[40]	Quercetin	-0.18056678771972656
[64]	Hydroxychloroquine	-0.2167699933052063
[74]	Melatonin	-0.22435645759105682
[84]	Methylprednisolone	-0.23988288640975952
```

In [34]:

```
len(clinical_drug_map)
```

Out[34]:

```
34
```

Non-oncology drugs pepurposing

Verely (TerraBio) found that large number of non-oncology drugs selectively inhibited subsets of cancer cell lines in a manner predictable from the cell lines’ molecular features - https://terra.bio/paper-spotlight-discovering-the-anticancer-potential-of-non-oncology-drugs/?utm_source=Terra+App&utm_campaign=018bbc2576-Mar-Apr22_newsletter&utm_medium=email&utm_term=0_ea2ec28eda-018bbc2576-1319556874&ct=t(03/01-04/30_2022)&mc_cid=018bbc2576&mc_eid=546f35b1cb (https://terra.bio/paper-spotlight-discovering-the-anticancer-potential-of-non-oncology-drugs/?utm_source=Terra+App&utm_campaign=018bbc2576-Mar-Apr22_newsletter&utm_medium=email&utm_term=0_ea2ec28eda-018bbc2576-1319556874&ct=t(03/01-04/30_2022)&mc_cid=018bbc2576&mc_eid=546f35b1cb)

In [8]:

```
from IPython.display import IFrame
IFrame(src="https://depmap.org/repurposing/", width='100%', height='400px')
```

Out[8]:

Home Stay informed

Discovering the anti-cancer potential of non
viability profilir

Corsello SM, Nagari RT, Spangler RD, Rossen J, Kocak M, Bryan JG, Humeidi R, Peck D,
Montgomery P, Ben-David U, Garvie CW, Chen Y, Rees MG, Lyons NJ, McFarland JM, W
Harrington CN, Greulich H, Meyerson M, Vazquez F, Subramanian A, Roth JA, Bi

bstract

nti-cancer uses of non-oncology drugs have occasionally been found, b
'e sought to create a public resource containing the growth inhibitory a

In [ ]:

```
```

In [ ]:

```
```

In [3]:

```python
from IPython.display import IFrame
IFrame(src="https://www.melatonin-research.net/index.php/MR/article/view/81#:~:text=These%20data%20support%20the%20observed,immunometabolic%20adjuvant%20for%20melanoma%20therapy.", width='100%', height='400px')
```

Out[3]:

Register Login {$loggedInUsername}

Melatonin Research

Current Archives Announcements About

Search

Home / Archives / Vol 3 No 2 (2020): Melatonin Research / Research Articles

Melatonin inhibits human melanoma cells proliferation and invasion via cell cycle arrest and cytoskeleton remodeling

Oncostatic effects of melatonin in melanoma cells

In [5]:

```
from IPython.display import IFrame
IFrame(src="https://www.webmd.com/melanoma-skin-cancer/news/20050711/curry-spice-fight-skin-cancer#:~:text=A%20new%20study%20shows%20that,treat%20form%20of%20skin%20cancer.", width='100%', height='400px')
```

Out[5]:

WebMD

MENU

Curry Spice May Fight Skin Cancer

Ingredient in Curry May Kill Melanoma Cells

By Jennifer Warner

In [7]:

```
from IPython.display import IFrame
IFrame(src="https://www.sciencedirect.com/science/article/pii/S075333222030175X", width='1
00%', height='400px')
```

Out[7]:

⚠ JavaScript is disabled on your browser. Please enable JavaScript to use all the features on this page.

ScienceDirect

Biomedicine & Pharmacotherapy
Volume 125, May 2020, 109984

Melanoma suppression by quercein is correlated with RIG-I and type I interferon signaling

Danhong Peng [a, b], Linjiao Chen [a, b], Yang Sun [b], Libo Sun [b], Qianqian Yin [b], Siyu Deng [b],

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

```

```

In [ ]:

```

```

In [156]:

```
ls
```

```
coronavirus-related-host-genes.tsv        Disease_clinical_trial_drugs.tsv
COVID19_clinical_trial_drugs.tsv          Disease_drug_repurposing.ipynb
COVID-19_drug_repurposing.ipynb           infer_drug.tsv
COVID-19_drug_repurposing_via_genes.ipynb Readme.md
covid19-host-genes.tsv
```

In [157]:

```
pwd
```

Out[157]:

```
'/home/jupyter/DRKG/drug_repurpose'
```

In [158]:

```
cd ..
```

```
/home/jupyter/DRKG
```

In [159]:

```
cd ..
```

```
/home/jupyter
```

In [160]:

```
ls
```

```
DRKG/  drkg-zip/  tutorials/  Untitled.ipynb
```

In [165]:

```
!sudo tar -czvf DRKG.tar.gz ./DRKG
```

```
./DRKG/
./DRKG/licenses/
./DRKG/licenses/Readme.md
./DRKG/LICENSE
./DRKG/.git/
./DRKG/.git/hooks/
./DRKG/.git/hooks/pre-applypatch.sample
./DRKG/.git/hooks/pre-receive.sample
./DRKG/.git/hooks/applypatch-msg.sample
./DRKG/.git/hooks/prepare-commit-msg.sample
./DRKG/.git/hooks/commit-msg.sample
./DRKG/.git/hooks/pre-commit.sample
./DRKG/.git/hooks/pre-push.sample
./DRKG/.git/hooks/pre-rebase.sample
./DRKG/.git/hooks/update.sample
./DRKG/.git/hooks/post-update.sample
./DRKG/.git/logs/
./DRKG/.git/logs/refs/
./DRKG/.git/logs/refs/remotes/
./DRKG/.git/logs/refs/remotes/origin/
./DRKG/.git/logs/refs/remotes/origin/HEAD
./DRKG/.git/logs/refs/heads/
./DRKG/.git/logs/refs/heads/master
./DRKG/.git/logs/HEAD
./DRKG/.git/refs/
./DRKG/.git/refs/remotes/
./DRKG/.git/refs/remotes/origin/
./DRKG/.git/refs/remotes/origin/HEAD
./DRKG/.git/refs/heads/
./DRKG/.git/refs/heads/master
./DRKG/.git/refs/tags/
./DRKG/.git/branches/
./DRKG/.git/objects/
./DRKG/.git/objects/info/
./DRKG/.git/objects/pack/
./DRKG/.git/objects/pack/pack-57e662c2925023cc905a79560e07dc85c822ece5.pack
./DRKG/.git/objects/pack/pack-57e662c2925023cc905a79560e07dc85c822ece5.idx
./DRKG/.git/config
./DRKG/.git/info/
./DRKG/.git/info/exclude
./DRKG/.git/HEAD
./DRKG/.git/packed-refs
./DRKG/.git/description
./DRKG/.git/index
./DRKG/connectivity.png
./DRKG/Readme.md
./DRKG/data/
./DRKG/data/embed/
./DRKG/data/embed/mol_edgepred.npy
./DRKG/data/embed/mol_infomax.npy
./DRKG/data/embed/entities.tsv
./DRKG/data/embed/._relations.tsv
./DRKG/data/embed/Readme.md
./DRKG/data/embed/mol_masking.npy
./DRKG/data/embed/relations.tsv
./DRKG/data/embed/DRKG_TransE_l2_relation.npy
./DRKG/data/embed/mol_contextpred.npy
```

```
./DRKG/data/embed/DRKG_TransE_l2_entity.npy
./DRKG/data/embed/._entities.tsv
./DRKG/data/drkg.tar.gz
./DRKG/data/._relation_glossary.tsv
./DRKG/data/._embed
./DRKG/data/entity2src.tsv
./DRKG/data/._entity2src.tsv
./DRKG/data/drkg.tsv
./DRKG/data/.ipynb_checkpoints/
./DRKG/data/drkg/
./DRKG/data/drkg/embed/
./DRKG/data/drkg/embed/mol_edgepred.npy
./DRKG/data/drkg/embed/mol_infomax.npy
./DRKG/data/drkg/embed/entities.tsv
./DRKG/data/drkg/embed/Readme.md
./DRKG/data/drkg/embed/mol_masking.npy
./DRKG/data/drkg/embed/relations.tsv
./DRKG/data/drkg/embed/DRKG_TransE_l2_relation.npy
./DRKG/data/drkg/embed/mol_contextpred.npy
./DRKG/data/drkg/embed/DRKG_TransE_l2_entity.npy
./DRKG/data/drkg/entity2src.tsv
./DRKG/data/drkg/drkg.tsv
./DRKG/data/drkg/.ipynb_checkpoints/
./DRKG/data/drkg/relation_glossary.tsv
./DRKG/data/._drkg.tsv
./DRKG/data/relation_glossary.tsv
./DRKG/utils/
./DRKG/utils/utils.py
./DRKG/utils/__pycache__/
./DRKG/utils/__pycache__/utils.cpython-37.pyc
./DRKG/embedding_analysis/
./DRKG/embedding_analysis/relation.eps
./DRKG/embedding_analysis/Train_embeddings.ipynb
./DRKG/embedding_analysis/Entity_similarity_analysis.ipynb
./DRKG/embedding_analysis/relation-sim.eps
./DRKG/embedding_analysis/Readme.md
./DRKG/embedding_analysis/.ipynb_checkpoints/
./DRKG/embedding_analysis/.ipynb_checkpoints/Edge_score_analysis-checkpoint.i
pynb
./DRKG/embedding_analysis/.ipynb_checkpoints/Train_embeddings-checkpoint.ipyn
b
./DRKG/embedding_analysis/.ipynb_checkpoints/Relation_similarity_analysis-che
ckpoint.ipynb
./DRKG/embedding_analysis/Relation_similarity_analysis.ipynb
./DRKG/embedding_analysis/Edge_score_analysis.ipynb
./DRKG/embedding_analysis/Edge_similarity_based_on_link_recommendation_result
s.ipynb
./DRKG/embedding_analysis/ckpts/
./DRKG/embedding_analysis/ckpts/TransE_l2_DRKG_0/
./DRKG/embedding_analysis/ckpts/TransE_l2_DRKG_1/
./DRKG/embedding_analysis/train/
./DRKG/embedding_analysis/train/drkg_valid.tsv
./DRKG/embedding_analysis/train/drkg_test.tsv
./DRKG/embedding_analysis/train/entities.tsv
./DRKG/embedding_analysis/train/relations.tsv
./DRKG/embedding_analysis/train/ten_fold/
./DRKG/embedding_analysis/train/ten_fold/part4.tsv
```

```
./DRKG/embedding_analysis/train/ten_fold/part1.tsv
./DRKG/embedding_analysis/train/ten_fold/part3.tsv
./DRKG/embedding_analysis/train/ten_fold/part9.tsv
./DRKG/embedding_analysis/train/ten_fold/part7.tsv
./DRKG/embedding_analysis/train/ten_fold/part0.tsv
./DRKG/embedding_analysis/train/ten_fold/part8.tsv
./DRKG/embedding_analysis/train/ten_fold/part6.tsv
./DRKG/embedding_analysis/train/ten_fold/part2.tsv
./DRKG/embedding_analysis/train/ten_fold/part5.tsv
./DRKG/embedding_analysis/train/drkg_train.tsv
./DRKG/.ipynb_checkpoints/
./DRKG/.ipynb_checkpoints/Readme-checkpoint.md
./DRKG/drkg_with_dgl/
./DRKG/drkg_with_dgl/loading_drkg_in_dgl.ipynb
./DRKG/drkg_with_dgl/Readme.md
./DRKG/drkg_with_dgl/.ipynb_checkpoints/
./DRKG/drkg_with_dgl/.ipynb_checkpoints/loading_drkg_in_dgl-checkpoint.ipynb
./DRKG/drugbank_info/
./DRKG/drugbank_info/drugbank_small_molecule.txt
./DRKG/drugbank_info/drugbank_smiles.txt
./DRKG/drugbank_info/drugbank_biotech.txt
./DRKG/drugbank_info/README.md
./DRKG/drugbank_info/drugbank_weight.txt
./DRKG/raw_graph_analysis/
./DRKG/raw_graph_analysis/edge_pair_jaccard_scores_sorted_overlap.tsv
./DRKG/raw_graph_analysis/edge_pair_jaccard_scores_sorted_jacard.tsv
./DRKG/raw_graph_analysis/Readme.md
./DRKG/raw_graph_analysis/Jaccard_scores_among_all_edge_types_in_DRKG.ipynb
./DRKG/raw_graph_analysis/.ipynb_checkpoints/
./DRKG/raw_graph_analysis/.ipynb_checkpoints/Jaccard_scores_among_all_edge_ty
pes_in_DRKG-checkpoint.ipynb
./DRKG/.gitignore
./DRKG/drug_repurpose/
./DRKG/drug_repurpose/COVID-19_drug_repurposing_via_genes.ipynb
./DRKG/drug_repurpose/covid19-host-genes.tsv
./DRKG/drug_repurpose/coronavirus-related-host-genes.tsv
./DRKG/drug_repurpose/COVID-19_drug_repurposing.ipynb
./DRKG/drug_repurpose/Readme.md
./DRKG/drug_repurpose/.ipynb_checkpoints/
./DRKG/drug_repurpose/.ipynb_checkpoints/COVID19_clinical_trial_drugs-checkpo
int.tsv
./DRKG/drug_repurpose/.ipynb_checkpoints/coronavirus-related-host-genes-check
point.tsv
./DRKG/drug_repurpose/.ipynb_checkpoints/Disease_drug_repurposing-checkpoint.
ipynb
./DRKG/drug_repurpose/.ipynb_checkpoints/COVID-19_drug_repurposing_via_genes-
checkpoint.ipynb
./DRKG/drug_repurpose/.ipynb_checkpoints/COVID-19_drug_repurposing-checkpoin
t.ipynb
./DRKG/drug_repurpose/.ipynb_checkpoints/Disease_clinical_trial_drugs-checkpo
int.tsv
./DRKG/drug_repurpose/COVID19_clinical_trial_drugs.tsv
./DRKG/drug_repurpose/Disease_drug_repurposing.ipynb
./DRKG/drug_repurpose/infer_drug.tsv
./DRKG/drug_repurpose/Disease_clinical_trial_drugs.tsv
```

In [166]:

```
ls
```

```
DRKG/  DRKG.tar.gz  drkg-zip/  tutorials/  Untitled.ipynb
```

In [167]:

```
!sudo tar -czvf drkg-zip.tar.gz ./drkg-zip
```

```
./drkg-zip/
./drkg-zip/embed/
./drkg-zip/embed/mol_edgepred.npy
./drkg-zip/embed/mol_infomax.npy
./drkg-zip/embed/entities.tsv
./drkg-zip/embed/._relations.tsv
./drkg-zip/embed/Readme.md
./drkg-zip/embed/mol_masking.npy
./drkg-zip/embed/relations.tsv
./drkg-zip/embed/DRKG_TransE_l2_relation.npy
./drkg-zip/embed/mol_contextpred.npy
./drkg-zip/embed/DRKG_TransE_l2_entity.npy
./drkg-zip/embed/._entities.tsv
./drkg-zip/drkg.tar
./drkg-zip/._relation_glossary.tsv
./drkg-zip/._embed
./drkg-zip/entity2src.tsv
./drkg-zip/._entity2src.tsv
./drkg-zip/drkg.tsv
./drkg-zip/._drkg.tsv
./drkg-zip/relation_glossary.tsv
```

In [ ]:

```

```