

# Seminar

Reporter: Lichuang Huang

Supervisor: Gang Cao

2022-04-08

1 Part 1: MCnebulia

2 part 2: pharmacology

3 END

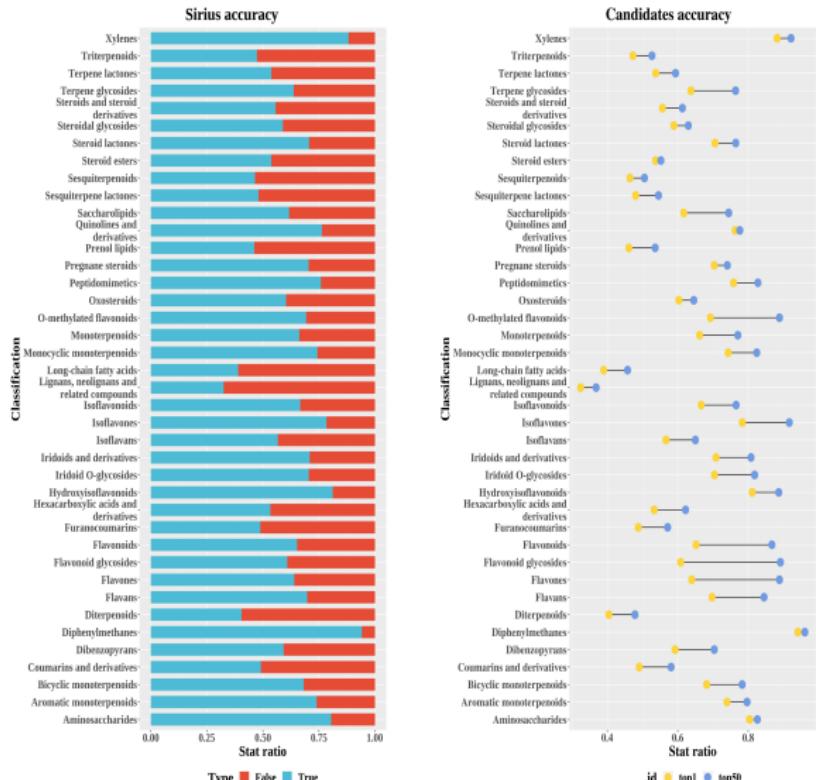
## Section 1

### **Part 1: MCnebula**

# Inovation

- ① Improve the compound prediction accuracy
  - via clustering in classification
  - via considering retention time
  - ...
- ② High accuracy classes clustering for MS/MS annotation
  - over 80% accuracy clustering, even unknown compound (no structure information)
- ③ Intuitive compound classes distribution in network visualization
  - each class involves a sub-nebula to explore the compound annotation
- ④ MCnebula algorithm integrated in R
  - cover SIRIUS LC-MS workflow analysis into R pipeline
- ⑤ A wide range of applicability
  - not be confined to metabolome identification
  - not be confined to spectrum library, but structure library

# Sirius accuracy



# Implement a new score for re-rank

Structure candidates re-rank score:

$$S_{re-rank} = (S_{simi} \times W_{simi}) + (S_{cluster} \times W_{cluster}) + (S_{RT} \times W_{RT})$$

# According to Retip

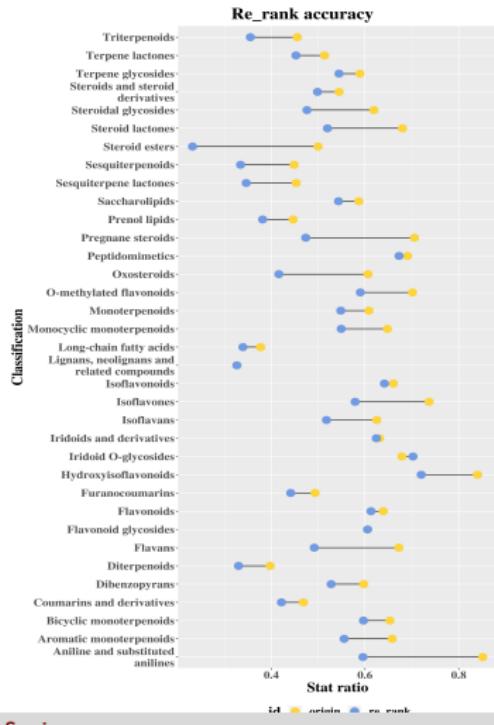
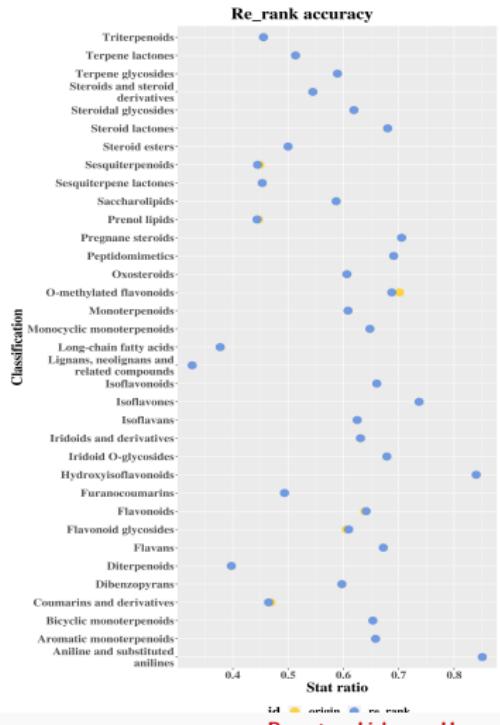
However, when I do this with `getCD(HILIC)`, it did failed.

```
[1] "Computing Chemical Descriptors 1 of 970 ... Please wait"  
  
*** caught segfault ***  
address 0x7ffda9d71fa8, cause 'memory not mapped'  
  
Traceback:  
1: .jcall(desc, "Lorg/openscience/cdk/qsar/DescriptorValue;", "calculate", a, check = check)  
2: tryCatchList(expr, classes, parentenv, handlers)  
3: tryCatch({ .jcall(desc, "Lorg/openscience/cdk/qsar/DescriptorValue;", "calculate", a, che  
4: FUN(X[[i]], ...)  
5: lapply(molecules, function(a, check) { val <- tryCatch({ .jcall(desc, "Lorg/openscience/cc  
6: rcdk::eval.desc(mols_x1, descNames)  
7: getCD(HILIC)
```

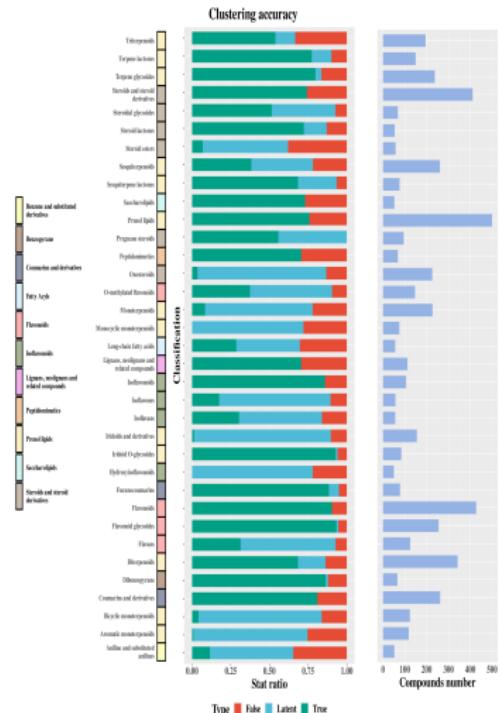
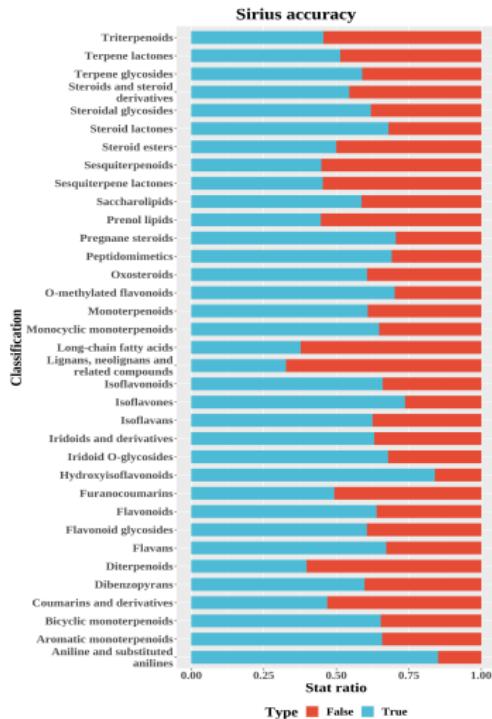
Figure 2: Retip issue

# Re-rank results

## Score re-rank and classification re-rank



# MCnebula accuracy



# Compare with GNPS and MolNetEnhancer

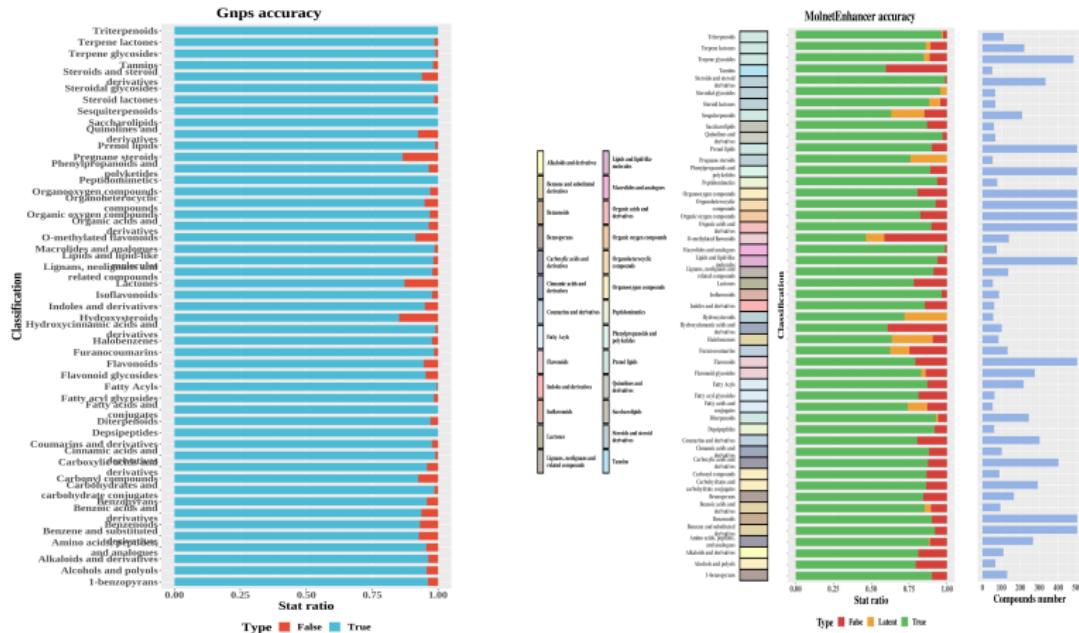


Figure 5: MolNetEnhancer

# COSMIC Noise design

- We simulated a global mass shift (bias) by drawing a random number  $\delta$  from  $\mathcal{N}(0, \sigma_{mb}^2)$  and then shifting every peak mass  $m$  by  $\delta m$ . The standard deviation  $\sigma_{mb}$  was chosen as  $\sigma_{mb} = (10/3) \times 10^{-6}$  (medium noise) or  $\sigma_{mb} = (15/3) \times 10^{-6}$  (high noise), so that the  $3\sigma_{mb}$  interval represents a 10-ppm shift for medium noise and a 15-ppm shift for high noise.
- We simulated individual mass deviations by drawing, for each peak with mass  $m$  individually, a random number  $\delta$  from  $\mathcal{N}(0, \sigma_{md}^2)$  and shifting the peak by  $\delta m$ . The standard deviation  $\sigma_{md}$  was chosen so that the  $3\sigma_{md}$  interval represents a 10-ppm shift for medium noise and a 20-ppm shift for high noise.
- We simulated intensity variations in the spectrum: each peak intensity was multiplied by an individual random number  $\epsilon$  drawn from  $\mathcal{N}(1, \sigma_{id}^2)$ . Variance was chosen as  $\sigma_{id}^2 = 1$  for medium noise and  $\sigma_{id}^2 = 2$  for high noise. Furthermore, 0.03 times the maximum peak intensity of the spectrum was subtracted from each peak intensity. If a peak intensity fell below the threshold of one thousands of the maximum intensity in the spectrum, the peak was discarded.
- Finally, we added ‘noise peaks’ to the spectrum. As uniformly choosing the mass of a noise peak would result in peaks that are too easy to spot and sort out by our subsequent analysis<sup>12</sup>, we, instead, used peaks that appeared in other measured spectra. In pre-processing, a pool of ‘noise peaks’ was gathered from the fragmentation spectra, using all peaks that did not have a molecular subformula decomposition of the known molecular formula of the precursor. For each spectrum,  $\alpha n$  of these ‘noise peaks’ were added to the spectrum, where  $n$  is the number of peaks in the spectrum and  $\alpha = 0.2$  for medium noise and  $\alpha = 0.4$  for high noise. Intensities of ‘noise peaks’ were adjusted for maximum peak intensities in the contributing and receiving spectrum.

## Add noise: main method

- Mass global shift
- Mass individual shift
- Intensity shift
- Add noise peak from other peak

## Section 2

**part 2: pharmacology**

# Kidney metabolome PCA (ion mode: neg)

The re-collect kidney metabolome dataset

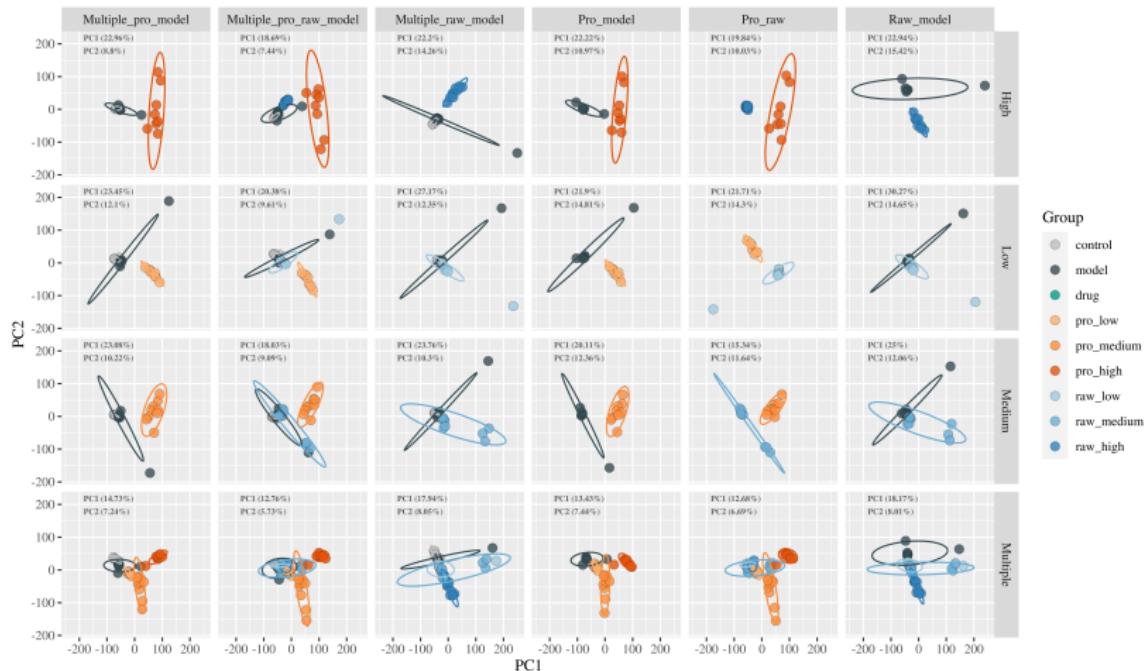
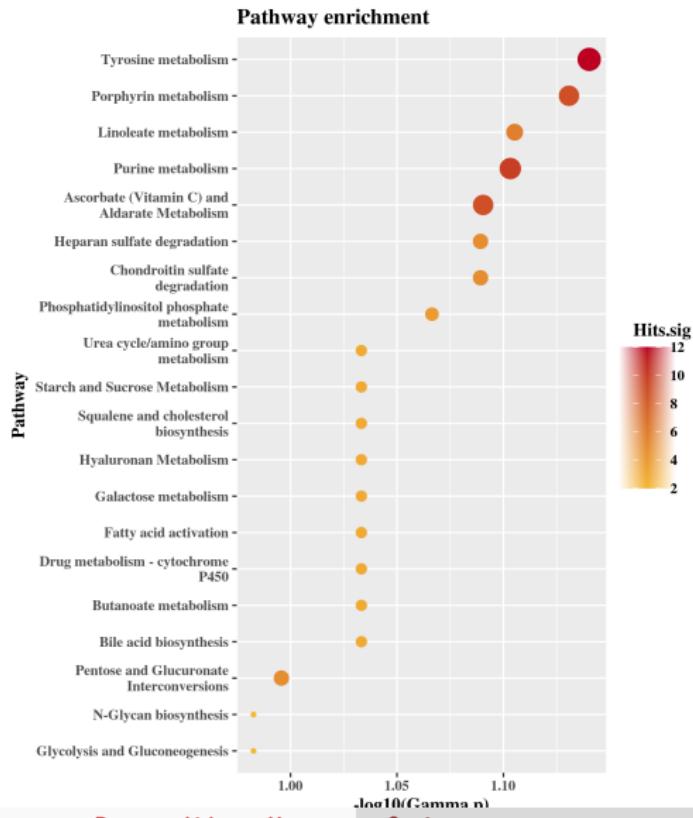


Figure 7: neg: kidney metabolome  
Reporter: Lichuang Huang  
Seminar

# MetaboAnalyst pathway enrichment



# Kidney metabolome PCA (ion mode: pos)

The re-collect kidney metabolome dataset

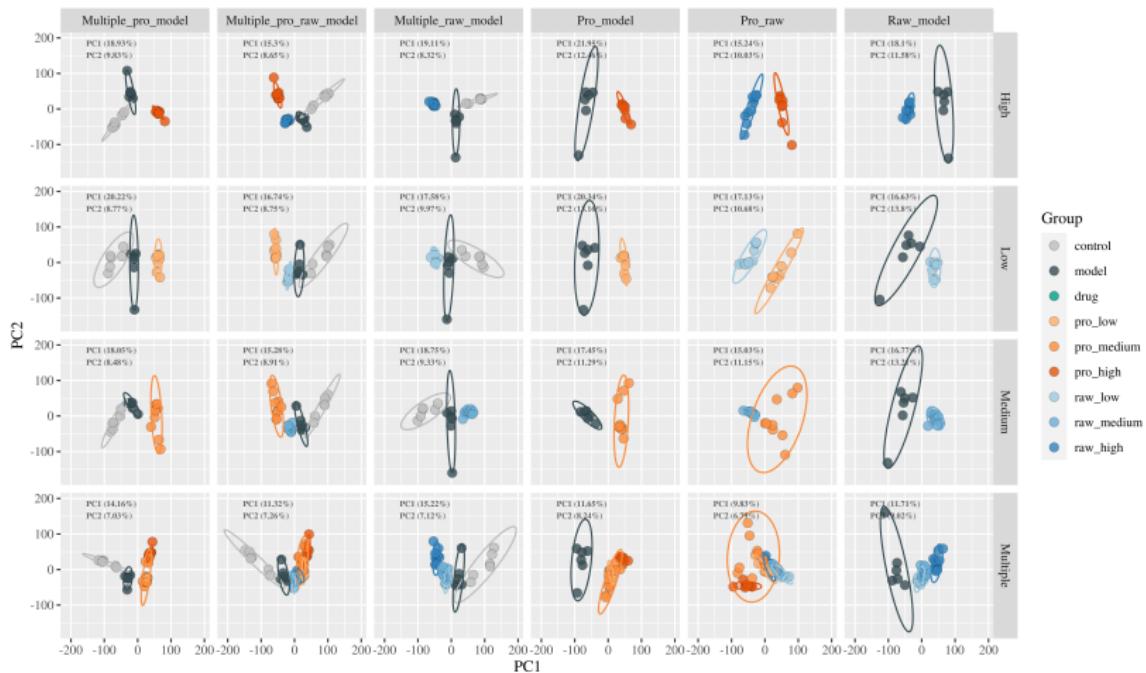


Figure 9: pos: kidney metabolome  
Reporter: Lichuang Huang  
Seminar

# Schedule

- MCnebula
  - ① Add noise
  - ② Parallel comparation (Qmistree MolNetEnhancer)
  - ③ Write Vignettes and manual reference document
  - ④ ...
- Renal LC-MS data analysis

## Section 3

**END**