

# NetRAX Experiment Evaluation

February 10, 2021

```
[1]: %matplotlib inline
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
sns.set(style="darkgrid")

[2]: def bic_stats(df):
    print("Inferred BIC better or equal: " + str(len(df[df['bic_inferred'] <=
    ↳df['bic_true']]))))
    print("Inferred BIC worse: " + str(len(df[df['bic_inferred'] >
    ↳df['bic_true']]))))
    df['bic_diff'].plot.hist(bins=100, alpha=0.5, title='(bic_true -
    ↳bic_inferred) / bic_true\n value >0 means inferred BIC was better')

def logl_stats(df):
    print("Inferred loglh better or equal: " + str(len(df[df['logl_inferred']
    ↳>= df['logl_true']]))))
    print("Inferred loglh worse: " + str(len(df[df['logl_inferred'] <
    ↳df['logl_true']]))))
    df['logl_diff'].plot.hist(bins=100, alpha=0.5, title='(logl_true -
    ↳logl_inferred) / logl_true\n value <0 means inferred logl was better')

def reticulation_stats(df):
    print("Inferred n_reticulations less: " +
    ↳str(len(df[df['n_reticulations_inferred'] < df['n_reticulations']]))))
    print("Inferred n_reticulations equal: " +
    ↳str(len(df[df['n_reticulations_inferred'] == df['n_reticulations']]))))
    print("Inferred n_reticulations more: " +
    ↳str(len(df[df['n_reticulations_inferred'] > df['n_reticulations']]))))

def weirdness_stats(df):
    df['true_network_weirdness'].plot.hist(bins=10, alpha=0.5, range=(0,1),
    ↳title='True network weirdness')

def zero_branches_stats(df):
    df['near_zero_branches_raxml'].plot.hist(bins=10, alpha=0.5,
    ↳title='Near-zero branches raxml')
```

```

def distances(df):
    fig, axes = plt.subplots(3, 2, constrained_layout=True)
    df['hardwired_cluster_distance'].plot.hist(bins=10, alpha=0.5,
    ↪title='Hardwired cluster distance', ax=axes[0,0])
    df['softwired_cluster_distance'].plot.hist(bins=10, alpha=0.5,
    ↪title='Softwired cluster distance', ax=axes[0,1])
    df['displayed_trees_distance'].plot.hist(bins=10, alpha=0.5,
    ↪title='Displayed trees distance', ax=axes[1,0])
    df['tripartition_distance'].plot.hist(bins=10, alpha=0.5,
    ↪title='Tripartition distance', ax=axes[1,1])
    df['nested_labels_distance'].plot.hist(bins=10, alpha=0.5, title='Nested_
    ↪labels distance', ax=axes[2,0])
    df['path_multiplicity_distance'].plot.hist(bins=10, alpha=0.5, title='Path_
    ↪multiplicity distance', ax=axes[2,1])

def build_stats(df):
    plt.figure(0)
    bic_stats(df)
    print("")
    plt.figure(1)
    logl_stats(df)
    print("")
    plt.figure(2)
    reticulation_stats(df)
    print("")
    plt.figure(3)
    weirdness_stats(df)
    print("")
    plt.figure(4)
    zero_branches_stats(df)
    print("")
    plt.figure(5)
    distances(df)

```

Load the result CSV:

```

[3]: #df = pd.read_csv('small_network_results.csv')
      #df = pd.read_csv('medium_network_norandom_results.csv')
      df = pd.read_csv('small_network_uniform_results.csv')
      #df = pd.read_csv('medium_network_norandom_uniform_results.csv')

      df['bic_diff'] = (df['bic_true'] - df['bic_inferred']) / df['bic_true']
      df['logl_diff'] = (df['logl_true'] - df['logl_inferred']) / df['logl_true']

```

```

[4]: pd.set_option('display.max_columns', None)
      df.head()

```

[4]:

```
name n_taxa n_trees \
0 datasets_small_network_uniform_0_0/0_5_taxa_1_... 5 2
1 datasets_small_network_uniform_0_0/0_5_taxa_1_... 5 2
2 datasets_small_network_uniform_0_0/0_5_taxa_1_... 5 2
3 datasets_small_network_uniform_0_0/0_5_taxa_1_... 5 2
4 datasets_small_network_uniform_0_0/0_5_taxa_1_... 5 2

n_reticulations msa_size sampling_type simulation_type \
0 1 100 PERFECT_SAMPLING CELINE
1 1 100 PERFECT_SAMPLING CELINE
2 1 100 PERFECT_SAMPLING CELINE
3 1 100 PERFECT_SAMPLING CELINE
4 1 200 PERFECT_SAMPLING CELINE

celine_params \
0 {'to': 0.21907766708685036| 'lambda': 14.56368...
1 {'to': 0.21907766708685036| 'lambda': 14.56368...
2 {'to': 0.21907766708685036| 'lambda': 14.56368...
3 {'to': 0.21907766708685036| 'lambda': 14.56368...
4 {'to': 0.21907766708685036| 'lambda': 14.56368...

seqgen_params near_zero_branches_raxml \
0 -mHKY -t3.0 -f0.3|0.2|0.2|0.3 0
1 -mHKY -t3.0 -f0.3|0.2|0.2|0.3 0
2 -mHKY -t3.0 -f0.3|0.2|0.2|0.3 0
3 -mHKY -t3.0 -f0.3|0.2|0.2|0.3 0
4 -mHKY -t3.0 -f0.3|0.2|0.2|0.3 0

n_equal_tree_pairs true_network_weirdness \
0 0 0
1 0 0
2 0 0
3 0 0
4 0 0

true_network_path \
0 datasets_small_network_uniform_0_0/0_5_taxa_1_...
1 datasets_small_network_uniform_0_0/0_5_taxa_1_...
2 datasets_small_network_uniform_0_0/0_5_taxa_1_...
3 datasets_small_network_uniform_0_0/0_5_taxa_1_...
4 datasets_small_network_uniform_0_0/0_5_taxa_1_...

inferred_network_path likelihood_type \
0 datasets_small_network_uniform_0_0/0_5_taxa_1_... AVERAGE
1 datasets_small_network_uniform_0_0/0_5_taxa_1_... AVERAGE
2 datasets_small_network_uniform_0_0/0_5_taxa_1_... BEST
3 datasets_small_network_uniform_0_0/0_5_taxa_1_... BEST
```

```

4 datasets_small_network_uniform_0_0/0_5_taxa_1_... AVERAGE

brlen_linkage_type start_type timeout n_random_start_networks \
0 LINKED FROM_RAXML 0 0
1 LINKED RANDOM 0 5
2 LINKED FROM_RAXML 0 0
3 LINKED RANDOM 0 5
4 LINKED FROM_RAXML 0 0

n_parsimony_start_networks runtime_inference n_reticulations_inferred \
0 0 116.881 0.0
1 5 727.941 0.0
2 0 62.698 0.0
3 5 245.541 0.0
4 0 109.374 0.0

bic_true logl_true bic_inferred logl_inferred bic_raxml \
0 819.124011 -316.342884 805.912651 -322.166420 805.889112
1 819.124011 -316.342884 805.884904 -322.152547 805.889112
2 819.124036 -316.342897 805.912651 -322.166420 805.889112
3 819.124036 -316.342897 805.884904 -322.152547 805.889112
4 1523.034144 -657.900743 1519.061342 -669.729852 1519.065015

logl_raxml rf_absolute_raxml rf_relative_raxml rf_absolute_inferred \
0 -322.154651 -1 -1 -1
1 -322.154651 -1 -1 -1
2 -322.154651 -1 -1 -1
3 -322.154651 -1 -1 -1
4 -669.731689 -1 -1 -1

rf_relative_inferred hardwired_cluster_distance \
0 -1 1.0
1 -1 2.0
2 -1 1.0
3 -1 2.0
4 -1 1.0

softwired_cluster_distance displayed_trees_distance \
0 1.0 1.5
1 2.0 1.5
2 1.0 1.5
3 2.0 1.5
4 1.0 1.5

tripartition_distance nested_labels_distance path_multiplicity_distance \
0 2.5 4.0 3.0
1 3.5 5.0 4.0

```

2	2.5	4.0	3.0
3	3.5	5.0	4.0
4	2.5	4.0	3.0

	bic_diff	logl_diff
0	0.016129	-0.018409
1	0.016163	-0.018365
2	0.016129	-0.018409
3	0.016163	-0.018365
4	0.002608	-0.017980

```
[5]: df.columns
```

```
[5]: Index(['name', 'n_taxa', 'n_trees', 'n_reticulations', 'msa_size',
         'sampling_type', 'simulation_type', 'celine_params', 'seqgen_params',
         'near_zero_branches_raxml', 'n_equal_tree_pairs',
         'true_network_weirdness', 'true_network_path', 'inferred_network_path',
         'likelihood_type', 'brlen_linkage_type', 'start_type', 'timeout',
         'n_random_start_networks', 'n_parsimony_start_networks',
         'runtime_inference', 'n_reticulations_inferred', 'bic_true',
         'logl_true', 'bic_inferred', 'logl_inferred', 'bic_raxml', 'logl_raxml',
         'rf_absolute_raxml', 'rf_relative_raxml', 'rf_absolute_inferred',
         'rf_relative_inferred', 'hardwired_cluster_distance',
         'softwired_cluster_distance', 'displayed_trees_distance',
         'tripartition_distance', 'nested_labels_distance',
         'path_multiplicity_distance', 'bic_diff', 'logl_diff'],
         dtype='object')
```

```
[6]: build_stats(df)
```

Inferred BIC better or equal: 979

Inferred BIC worse: 45

Inferred loglh better or equal: 370

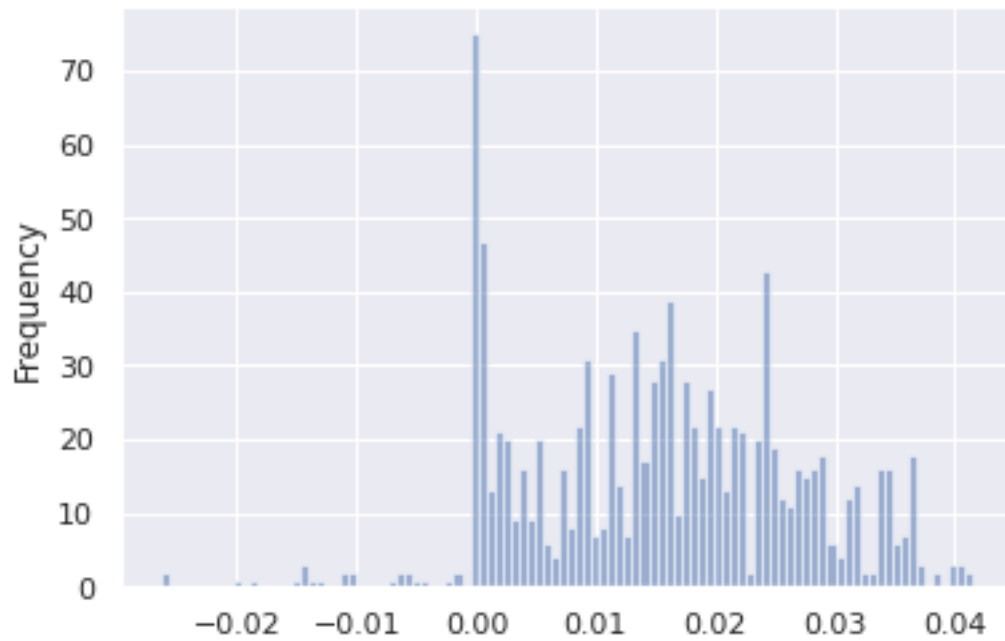
Inferred loglh worse: 654

Inferred n\_reticulations less: 872

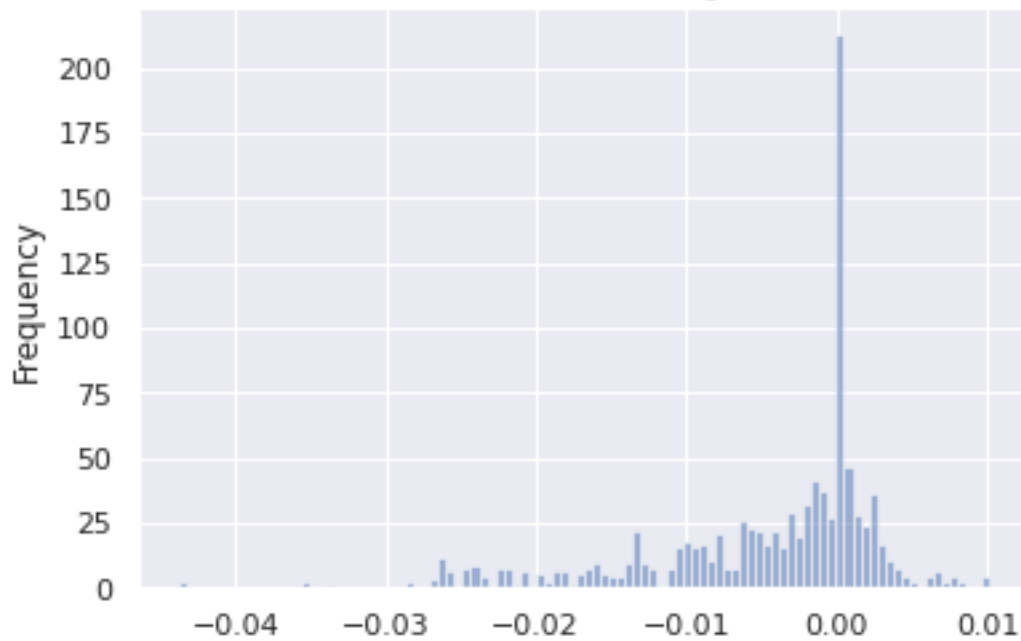
Inferred n\_reticulations equal: 152

Inferred n\_reticulations more: 0

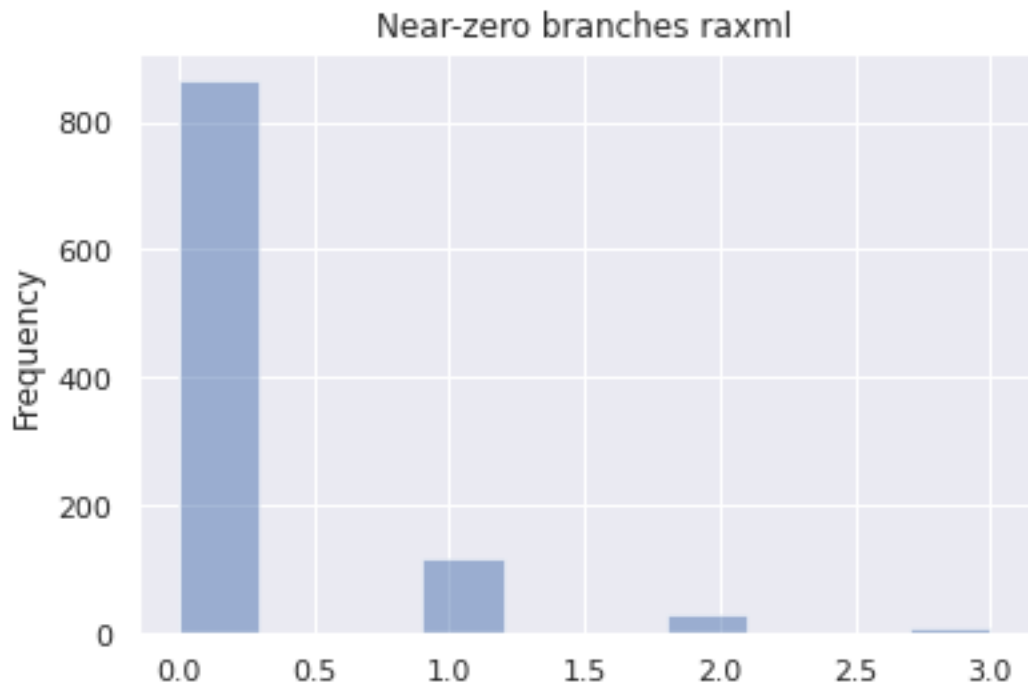
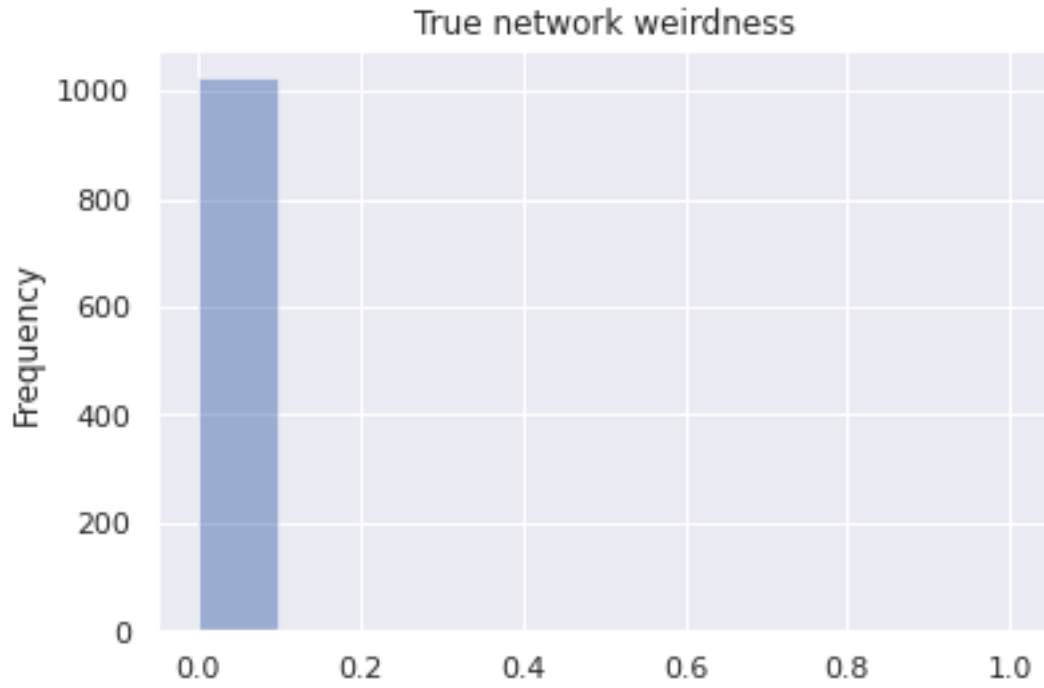
$(bic\_true - bic\_inferred) / bic\_true$   
value >0 means inferred BIC was better



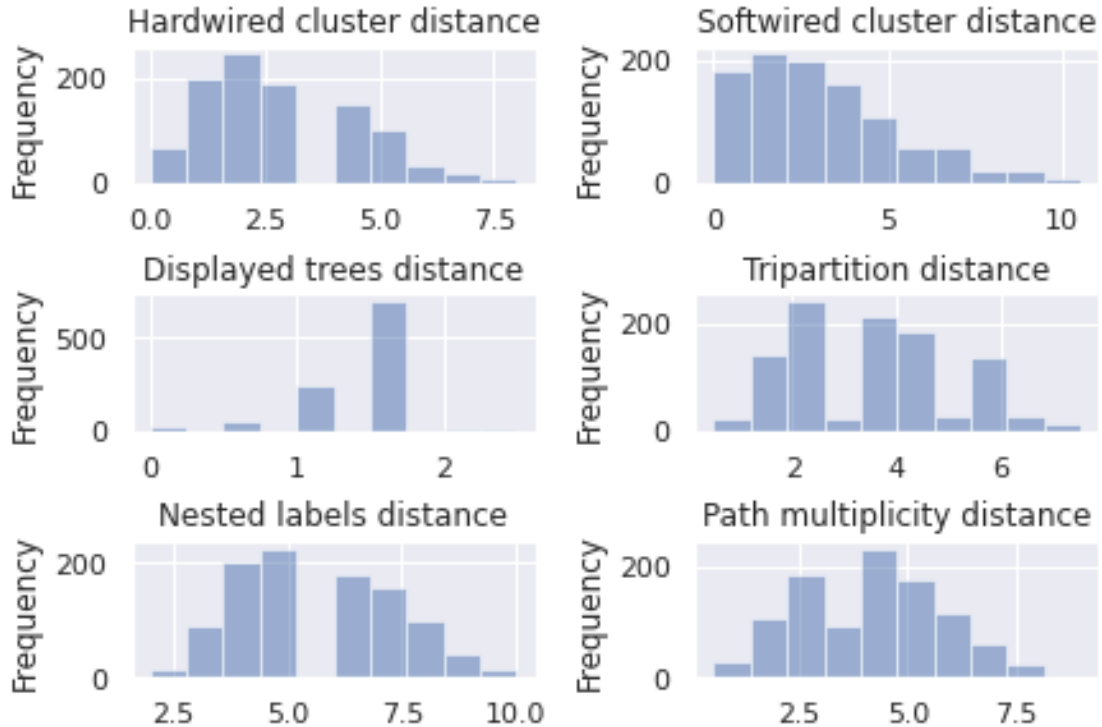
$(logl\_true - logl\_inferred) / logl\_true$   
value <0 means inferred logl was better



<Figure size 432x288 with 0 Axes>



<Figure size 432x288 with 0 Axes>



## 1 Plots for starting with raxml-ng best tree only

```
[7]: df_raxml_only = df.query('start_type == "FROM_RAXML"')
      build_stats(df_raxml_only)
```

Inferred BIC better or equal: 474

Inferred BIC worse: 38

Inferred loglh better or equal: 169

Inferred loglh worse: 343

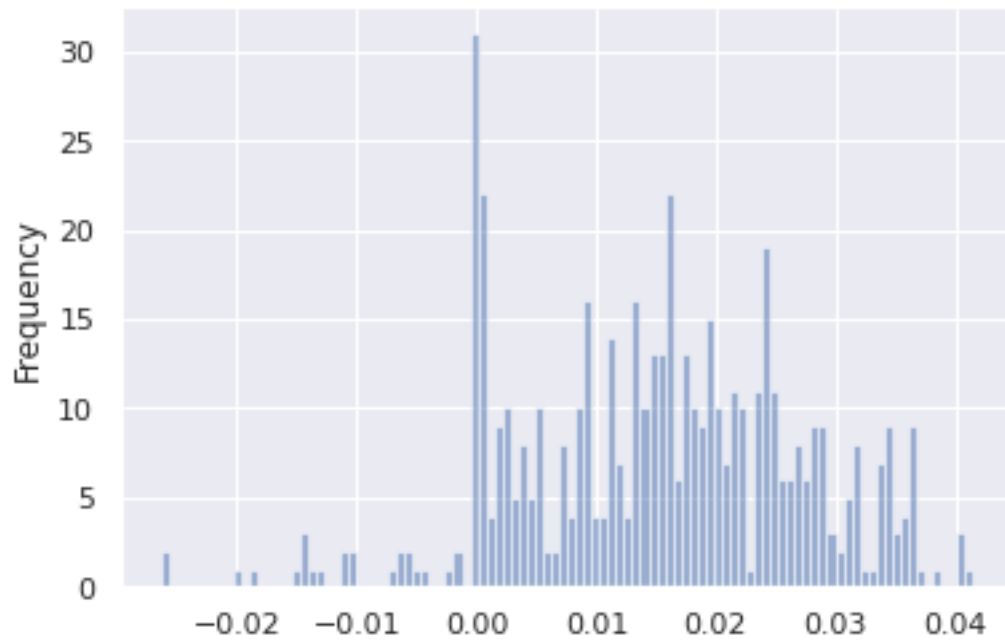
Inferred n\_reticulations less: 444

Inferred n\_reticulations equal: 68

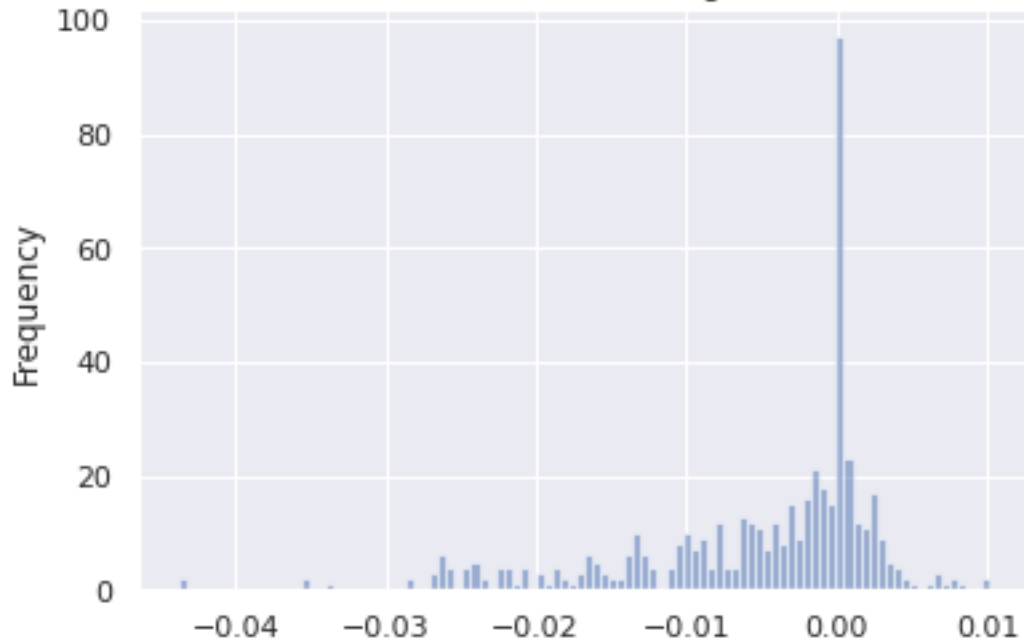
Inferred n\_reticulations more: 0



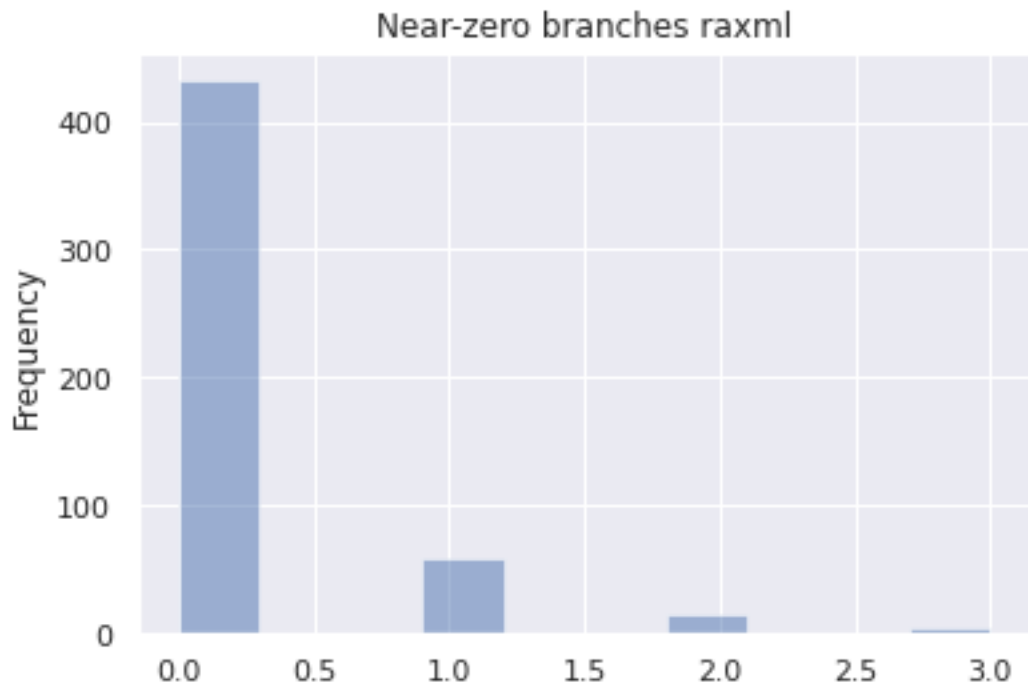
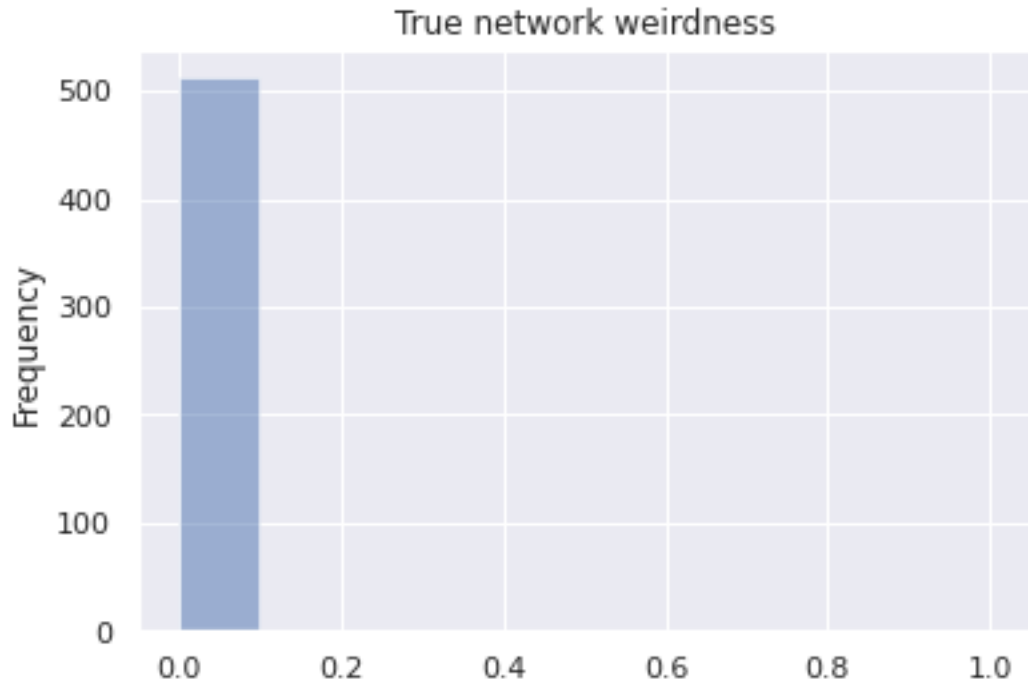
$(bic\_true - bic\_inferred) / bic\_true$   
value >0 means inferred BIC was better



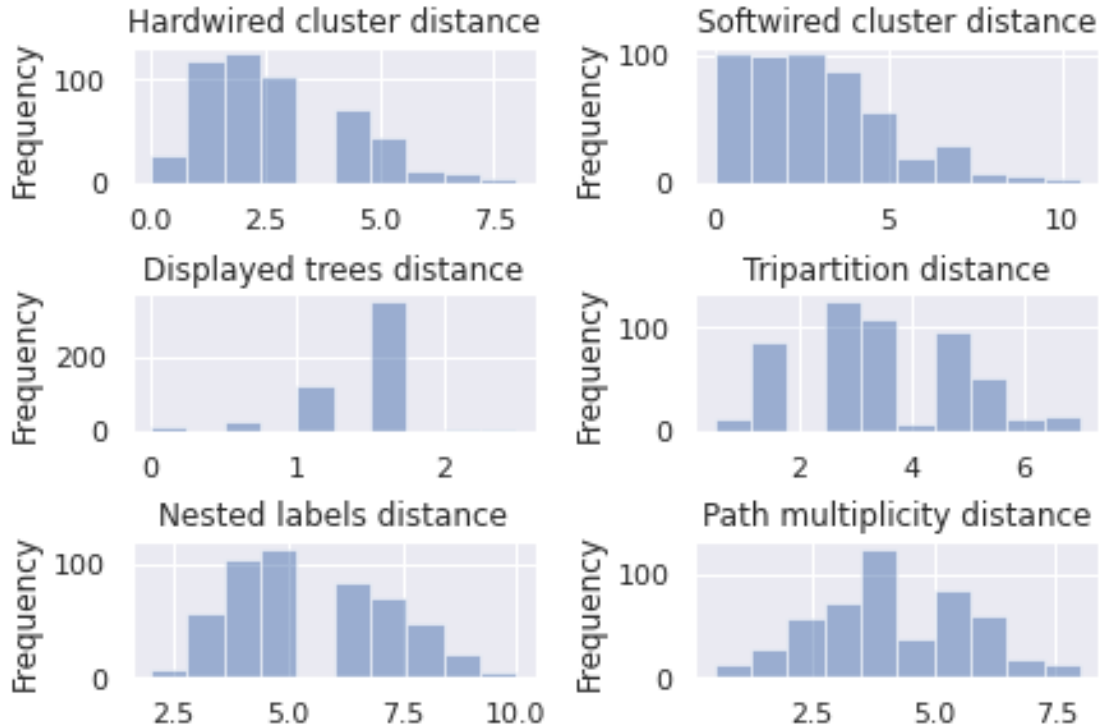
$(logl\_true - logl\_inferred) / logl\_true$   
value <0 means inferred logl was better



<Figure size 432x288 with 0 Axes>



<Figure size 432x288 with 0 Axes>



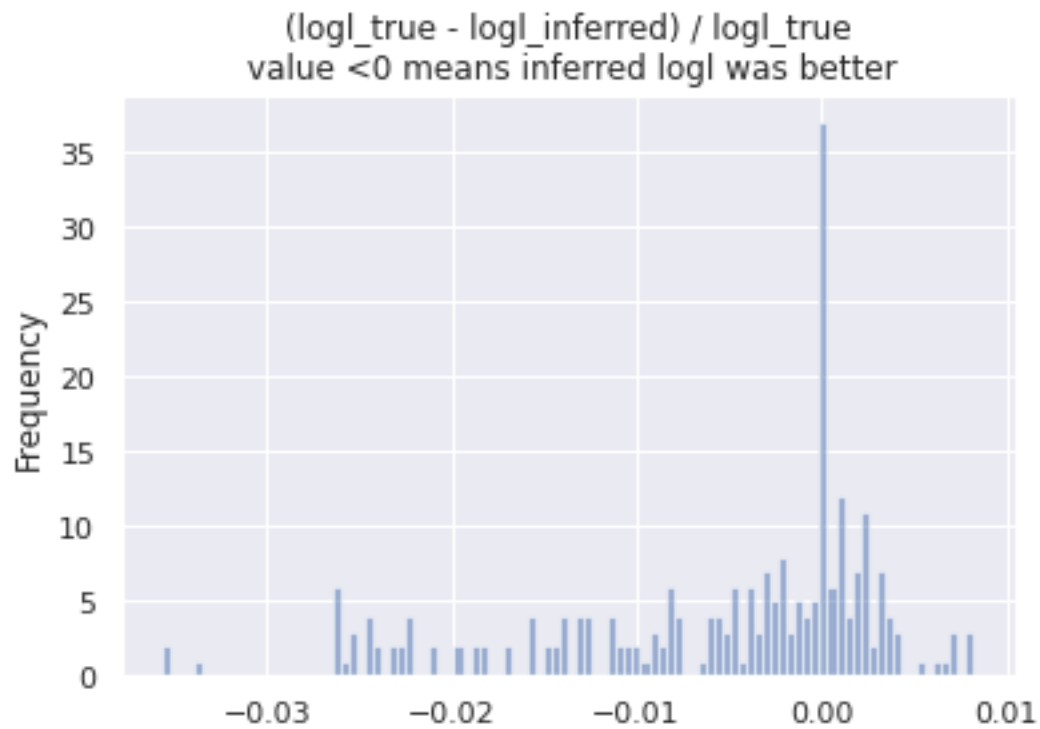
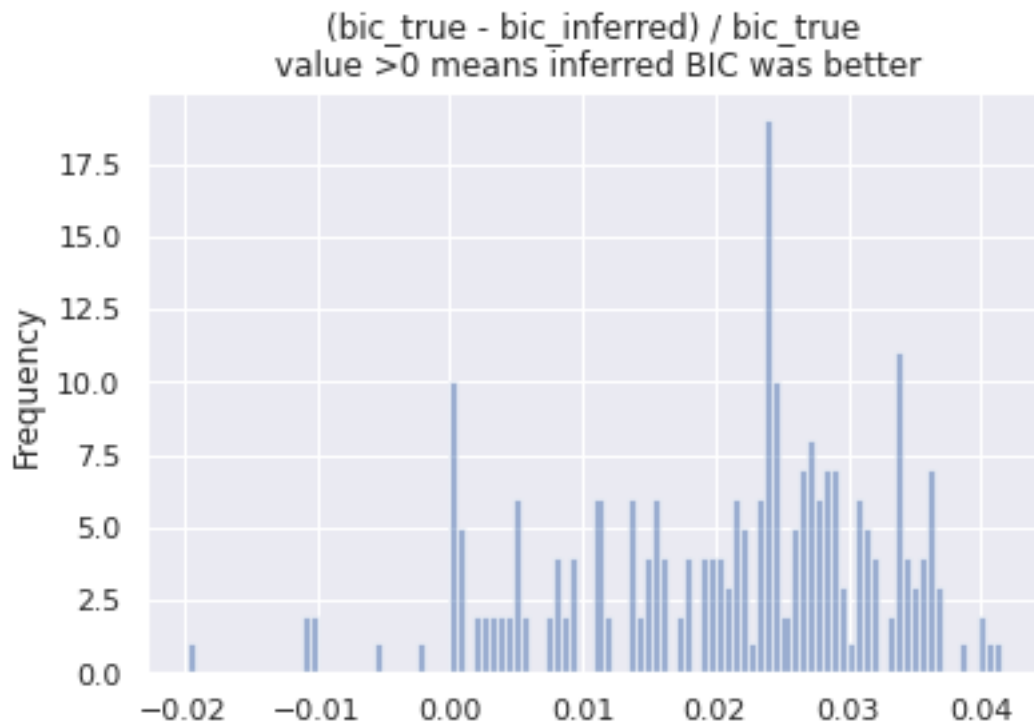
### 1.1 Plots for MSA\_size ~ 100\*n\_trees

```
[8]: df_raxml_only_msasize_100 = df_raxml_only.query('msa_size == 100')
      build_stats(df_raxml_only_msasize_100)
```

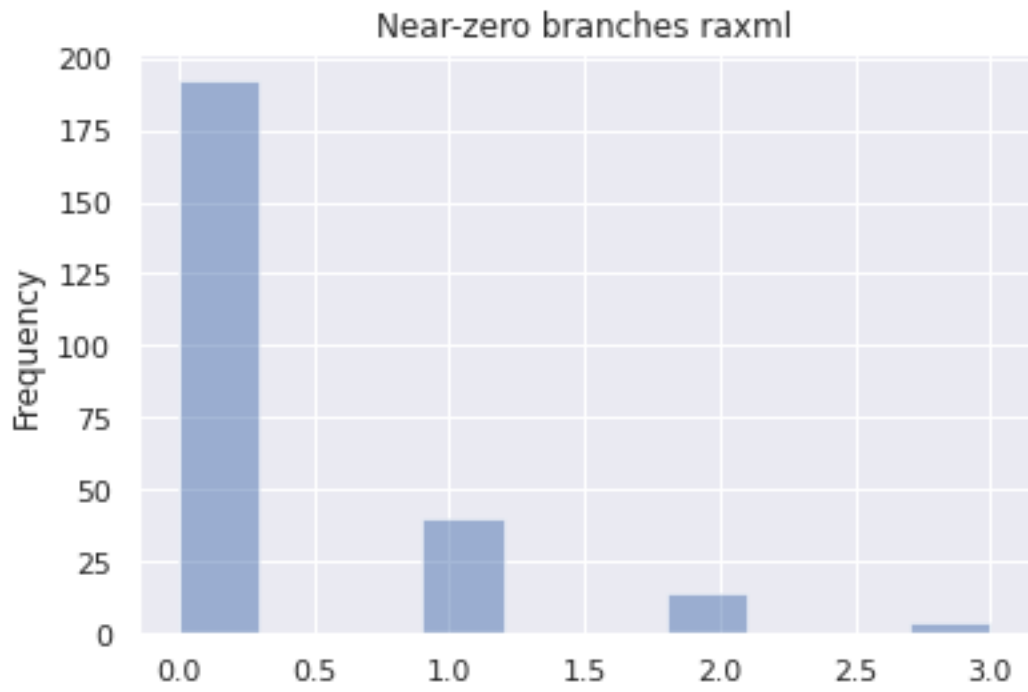
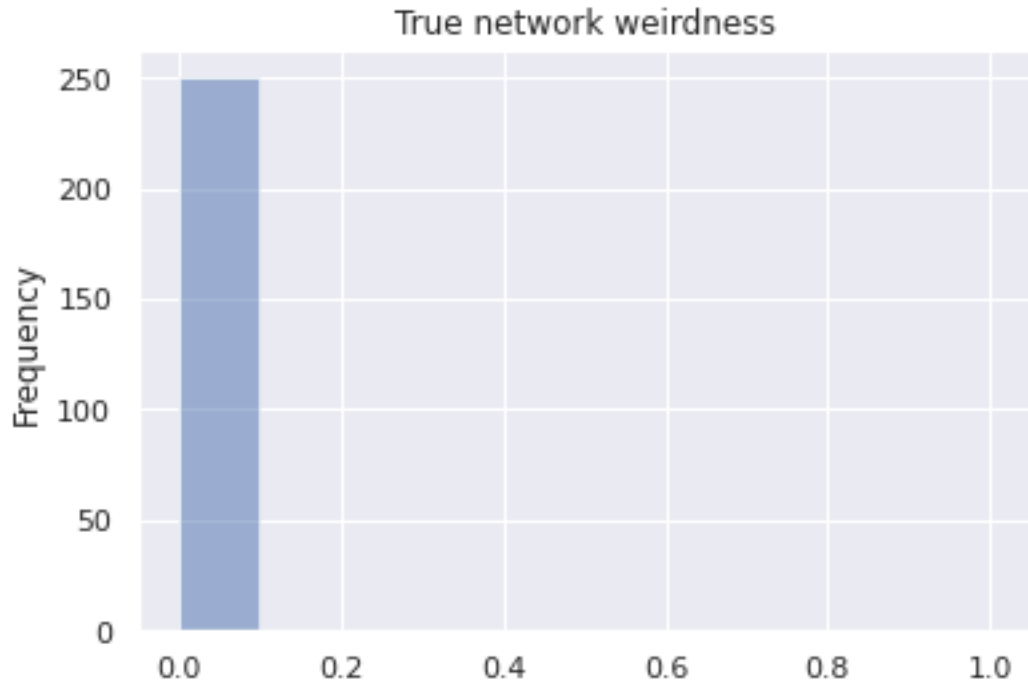
Inferred BIC better or equal: 240  
Inferred BIC worse: 10

Inferred loglh better or equal: 90  
Inferred loglh worse: 160

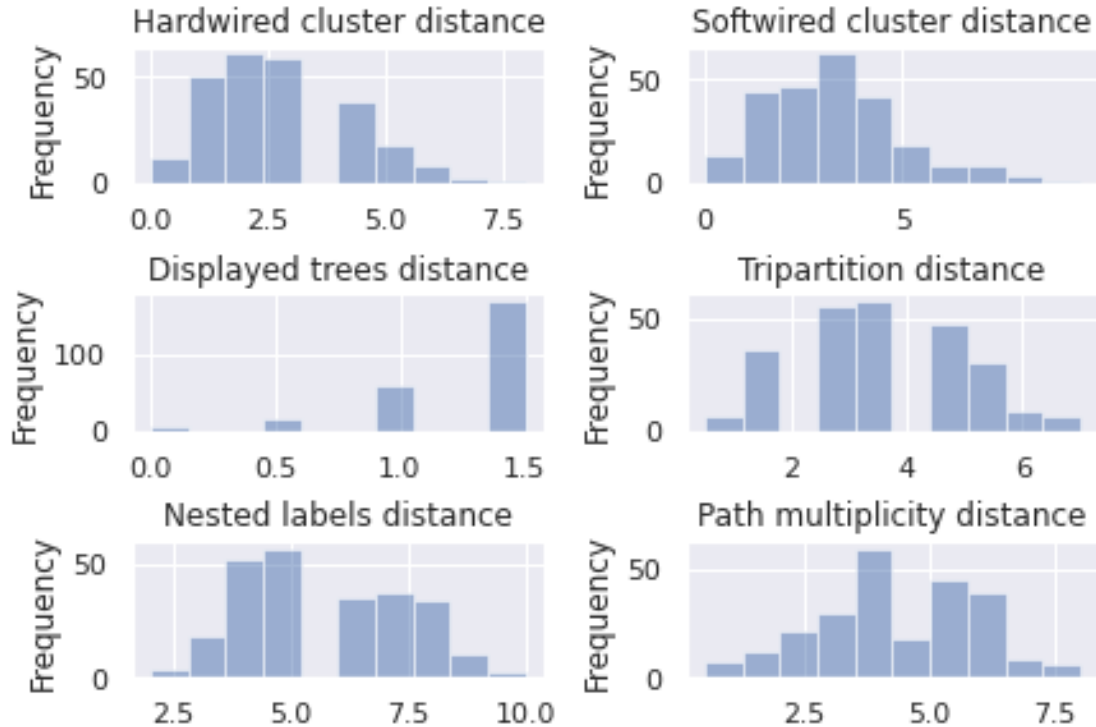
Inferred n\_reticulations less: 228  
Inferred n\_reticulations equal: 22  
Inferred n\_reticulations more: 0



<Figure size 432x288 with 0 Axes>



<Figure size 432x288 with 0 Axes>



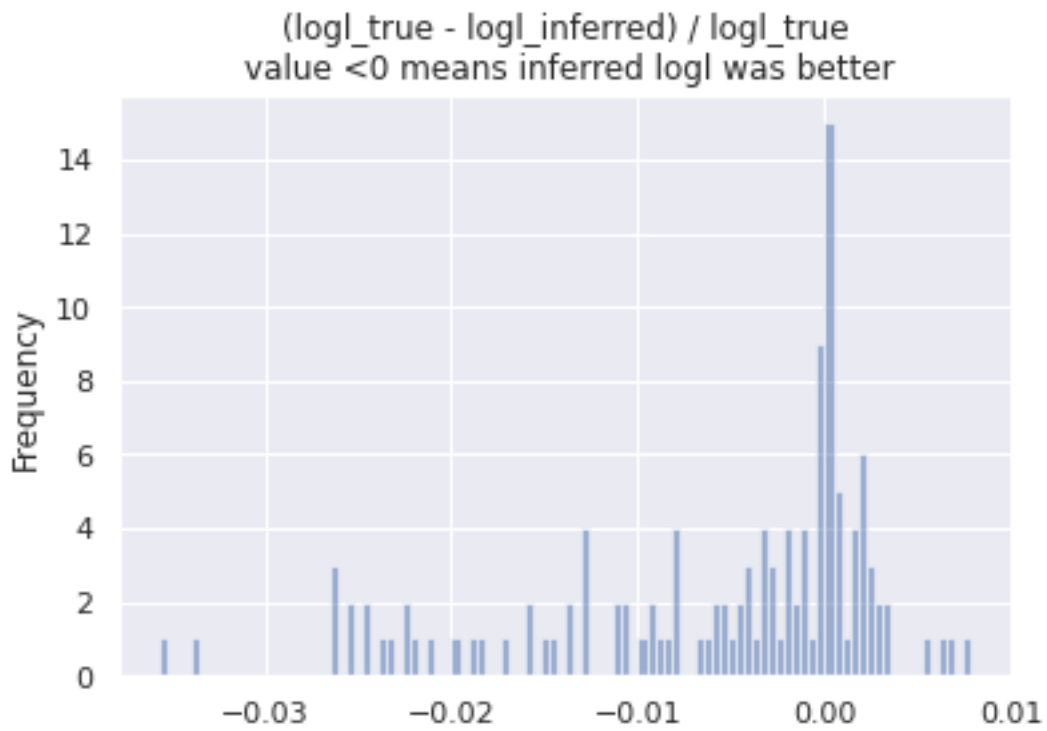
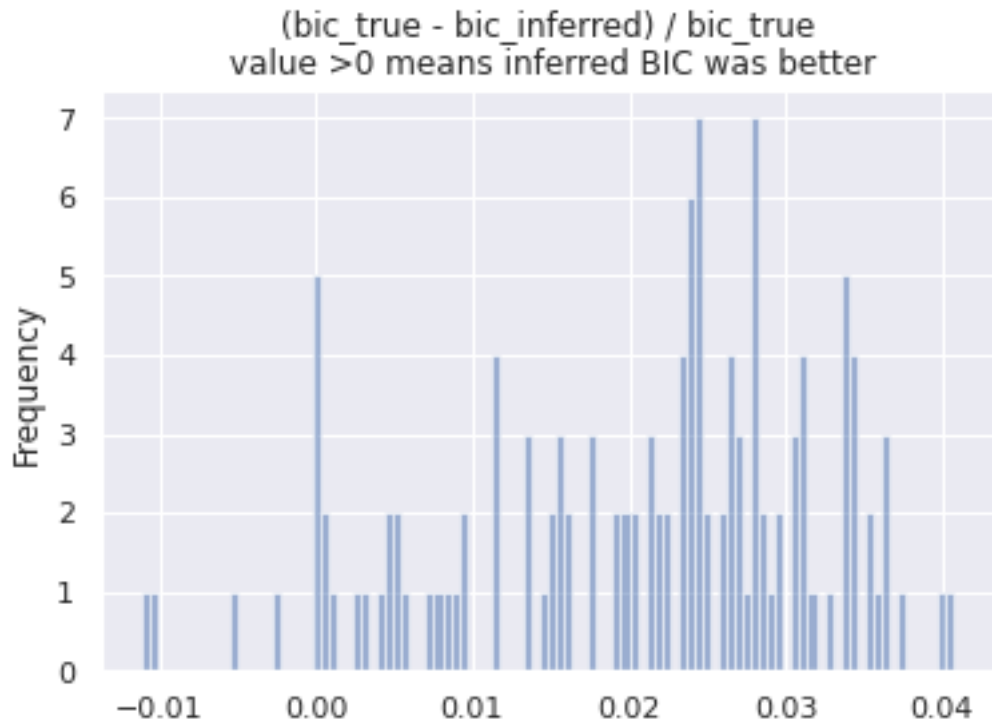
### 1.1.1 Plots for LikelihoodType.AVERAGE

```
[9]: df_raxml_only_msasize_100_average = df_raxml_only_msasize_100.  
      ↪query('likelihood_type == "AVERAGE"')  
      build_stats(df_raxml_only_msasize_100_average)
```

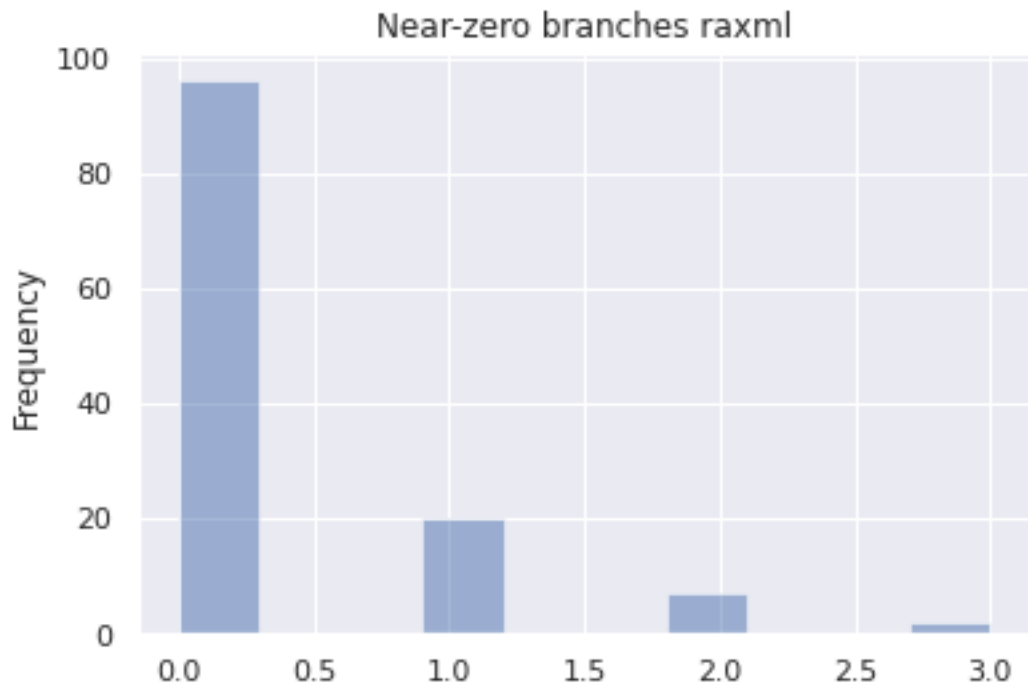
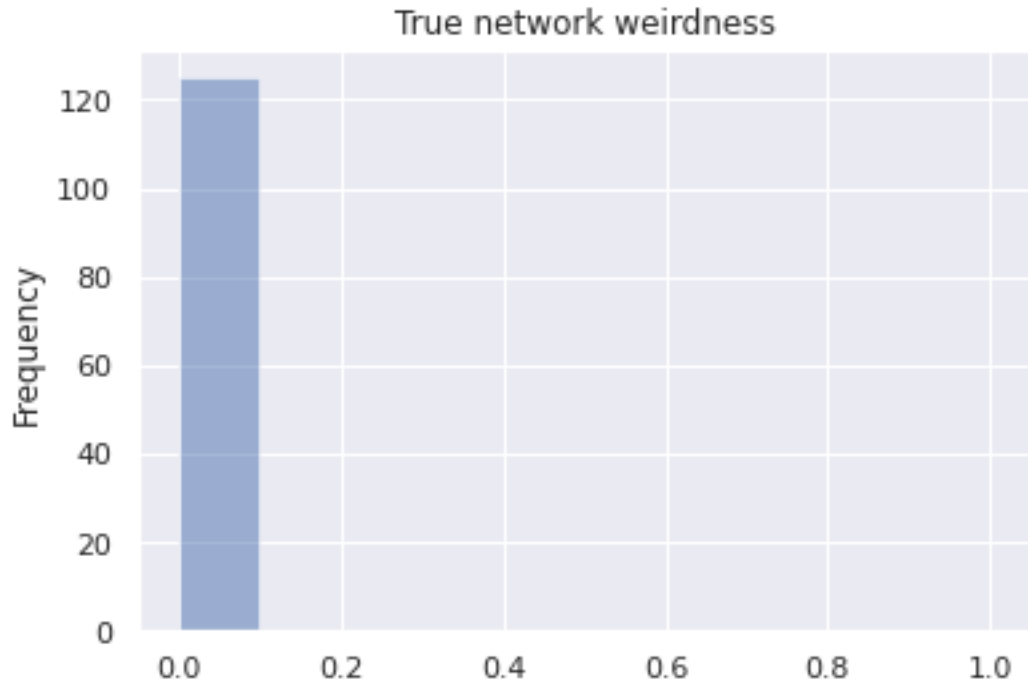
Inferred BIC better or equal: 121  
Inferred BIC worse: 4

Inferred loglh better or equal: 42  
Inferred loglh worse: 83

Inferred n\_reticulations less: 115  
Inferred n\_reticulations equal: 10  
Inferred n\_reticulations more: 0

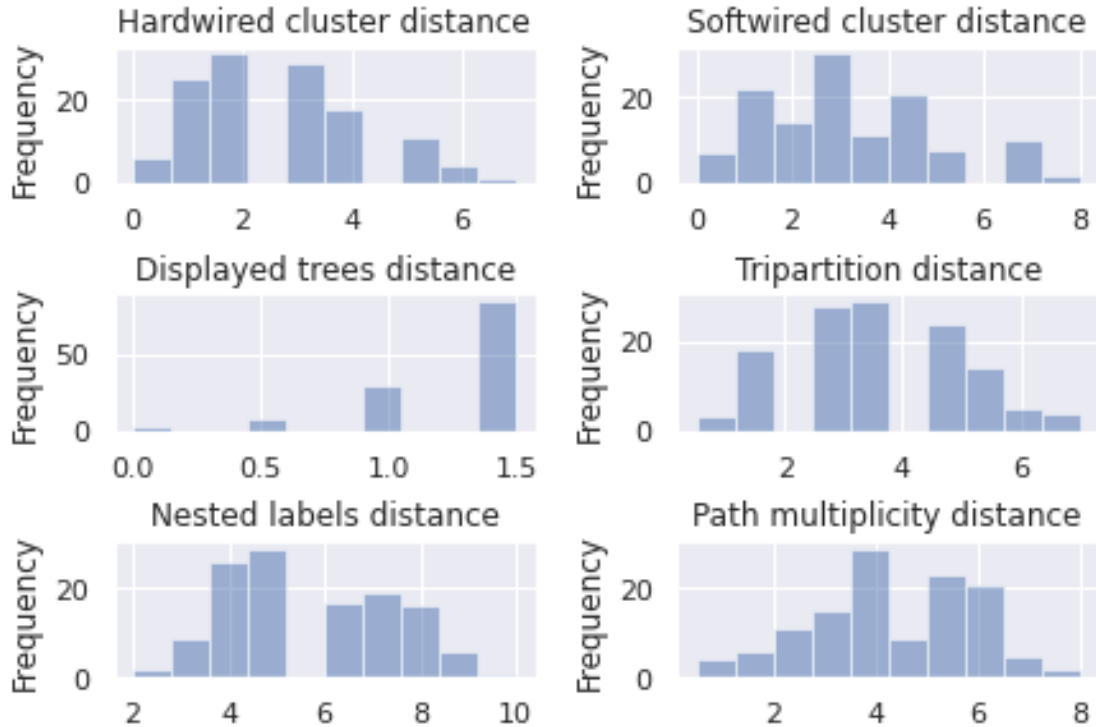


<Figure size 432x288 with 0 Axes>





<Figure size 432x288 with 0 Axes>



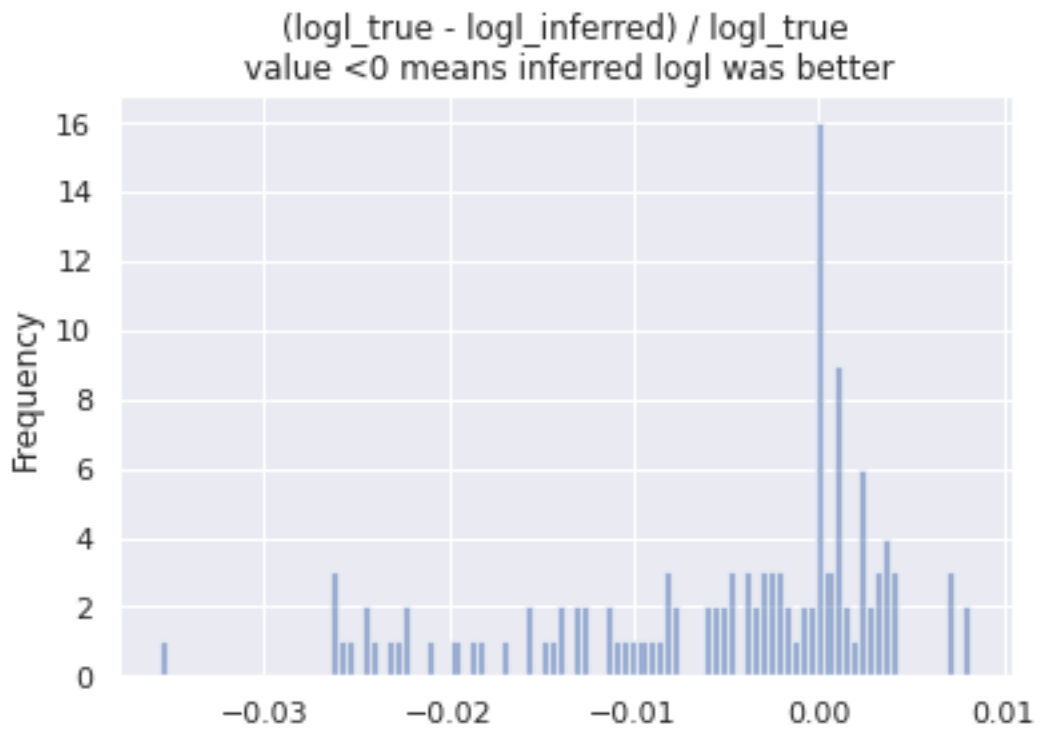
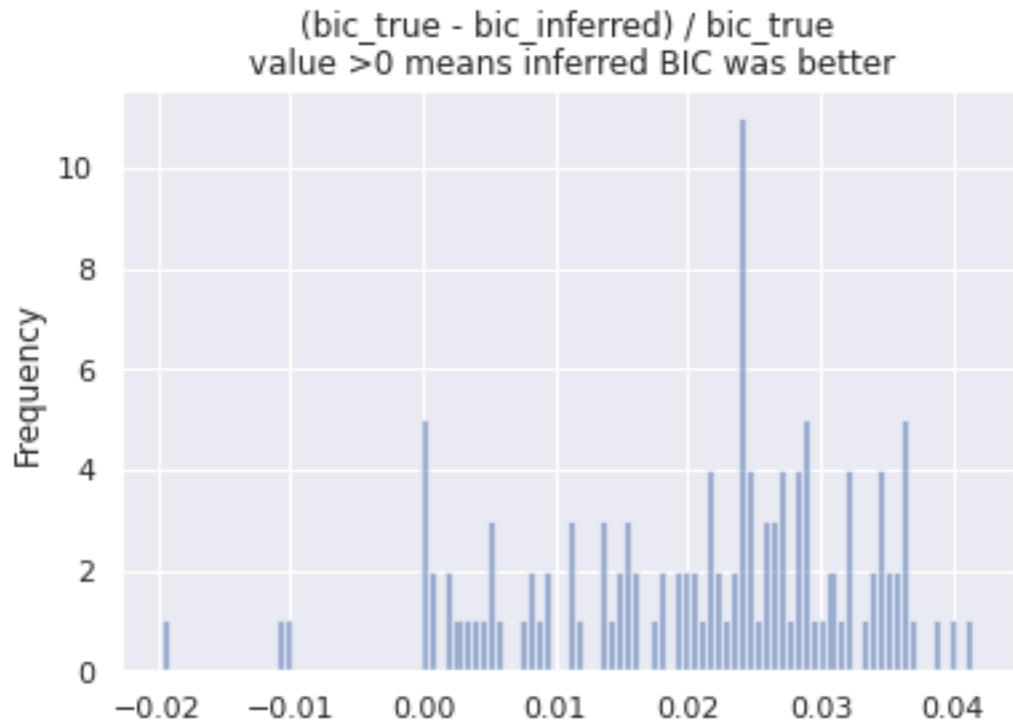
### 1.1.2 Plots for LikelihoodType.BEST

```
[10]: df_raxml_only_msasize_100_best = df_raxml_only_msasize_100.  
      ↪query('likelihood_type == "BEST"')  
      build_stats(df_raxml_only_msasize_100_best)
```

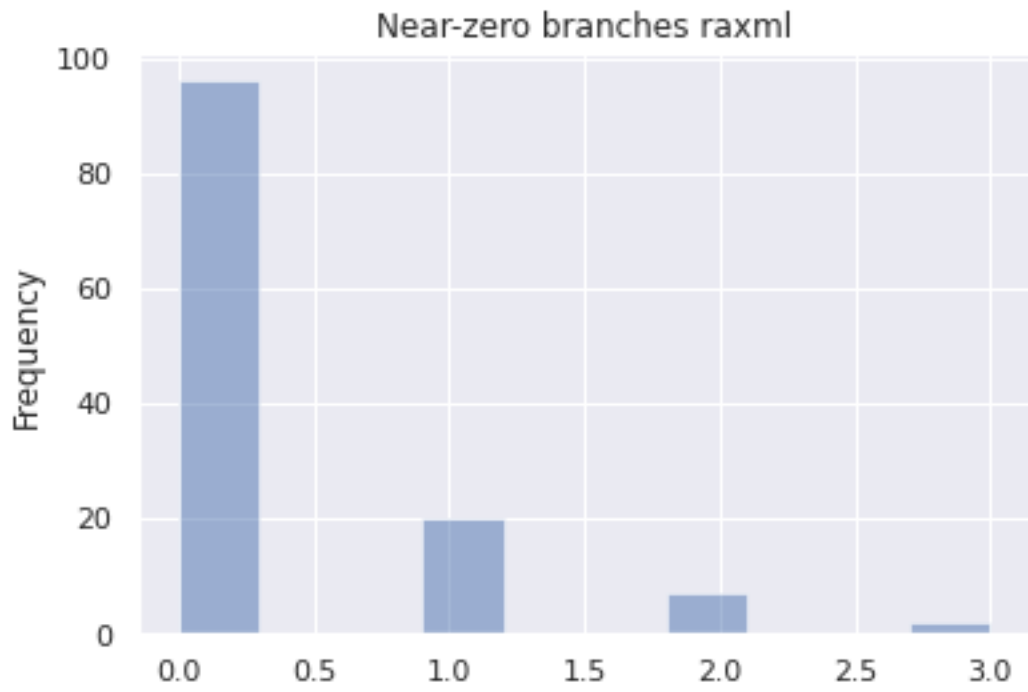
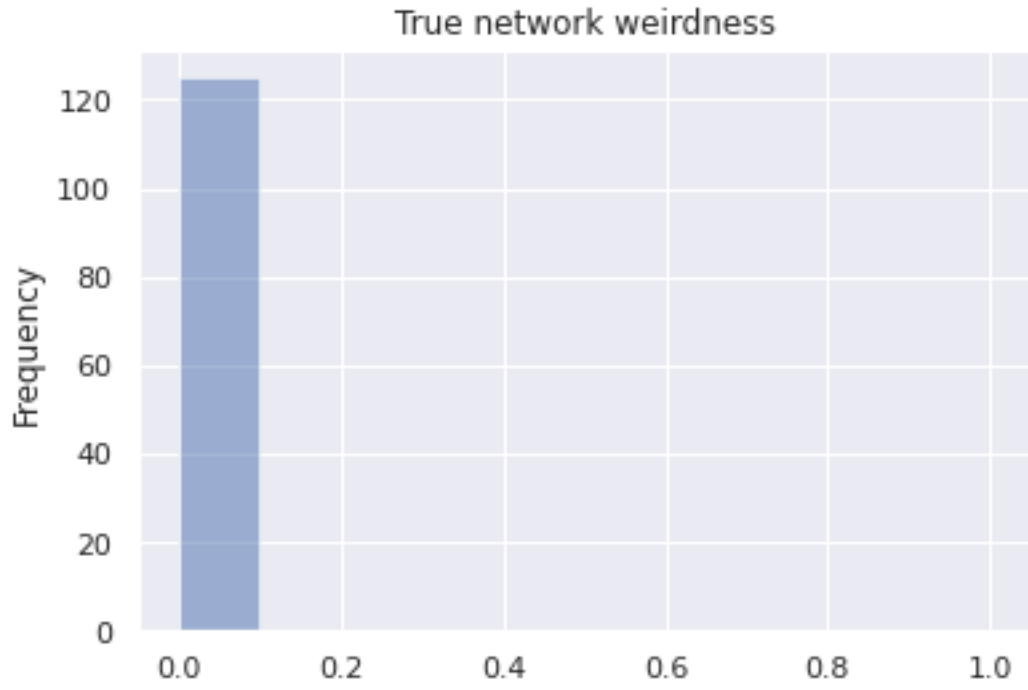
Inferred BIC better or equal: 119  
Inferred BIC worse: 6

Inferred loglh better or equal: 48  
Inferred loglh worse: 77

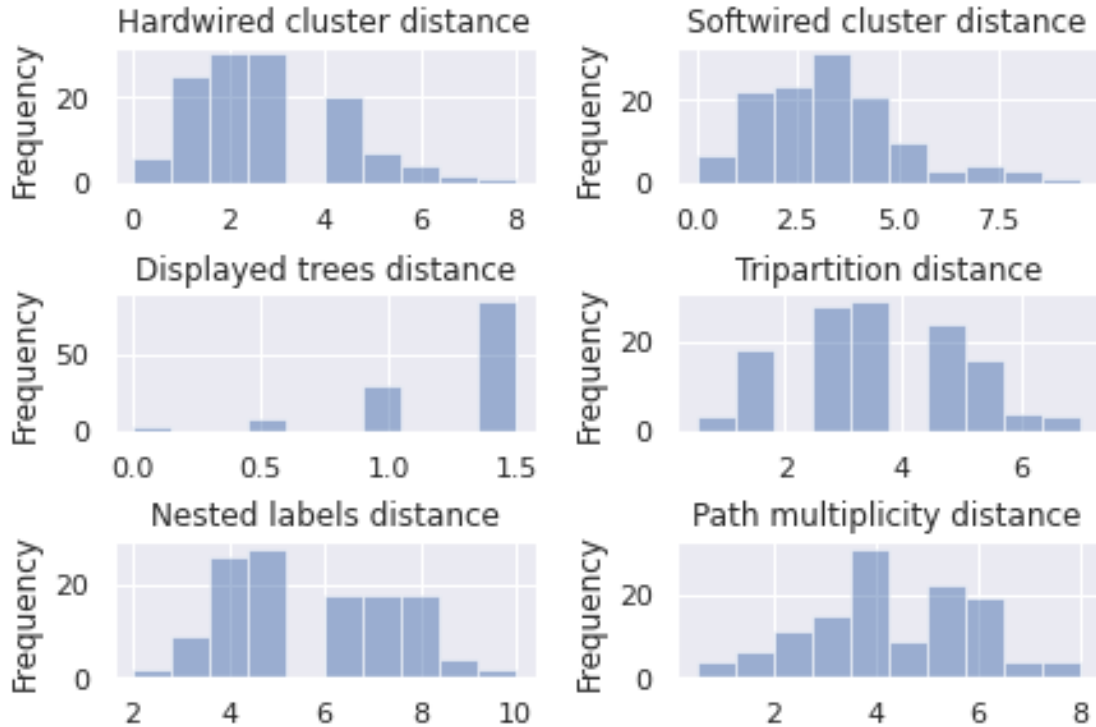
Inferred n\_reticulations less: 113  
Inferred n\_reticulations equal: 12  
Inferred n\_reticulations more: 0



<Figure size 432x288 with 0 Axes>



<Figure size 432x288 with 0 Axes>



## 1.2 Plots for MSA\_size ~ 200\*n\_trees

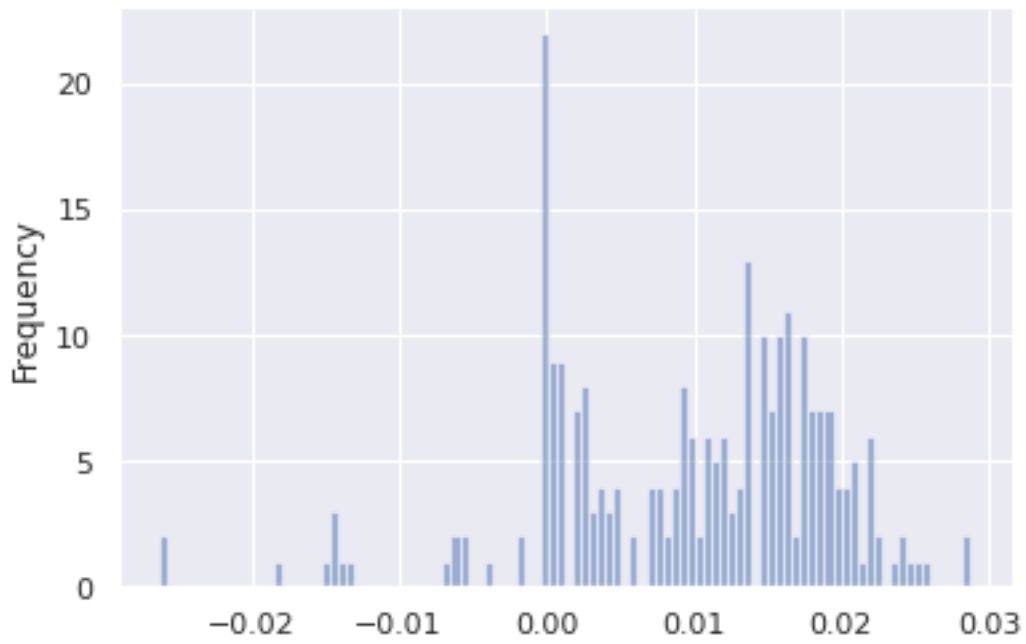
```
[11]: df_raxml_only_msasize_200 = df_raxml_only.query('msa_size == 200')  
      build_stats(df_raxml_only_msasize_200)
```

Inferred BIC better or equal: 230  
Inferred BIC worse: 26

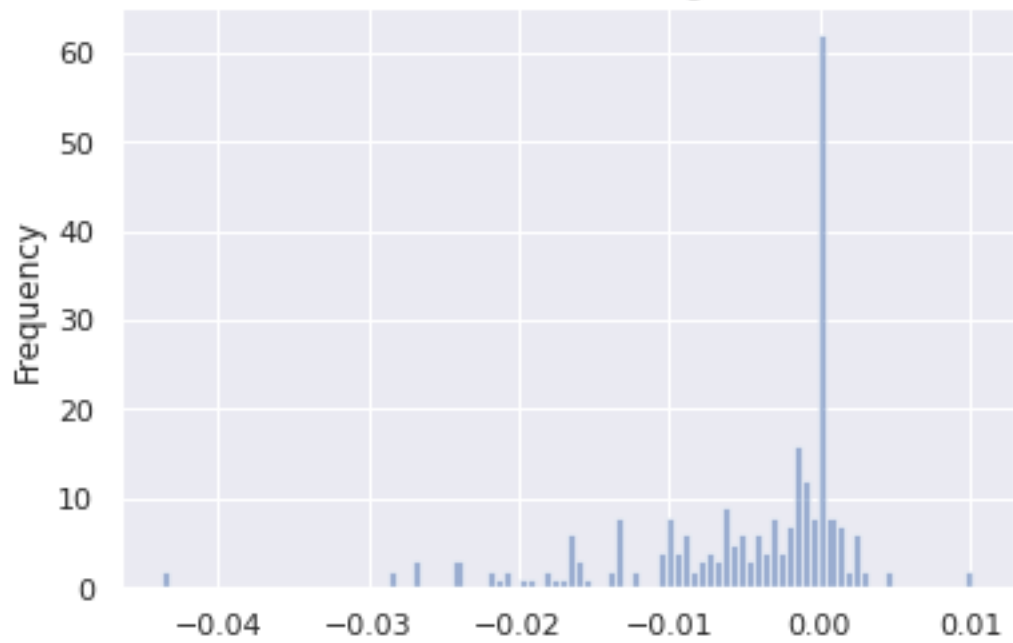
Inferred loglh better or equal: 79  
Inferred loglh worse: 177

Inferred n\_reticulations less: 210  
Inferred n\_reticulations equal: 46  
Inferred n\_reticulations more: 0

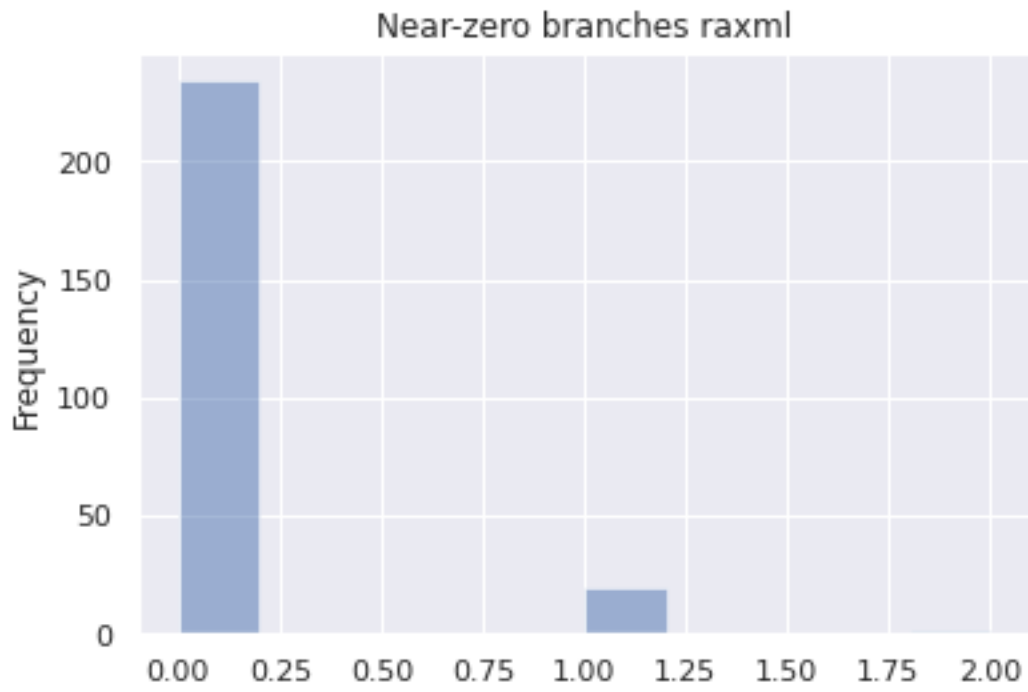
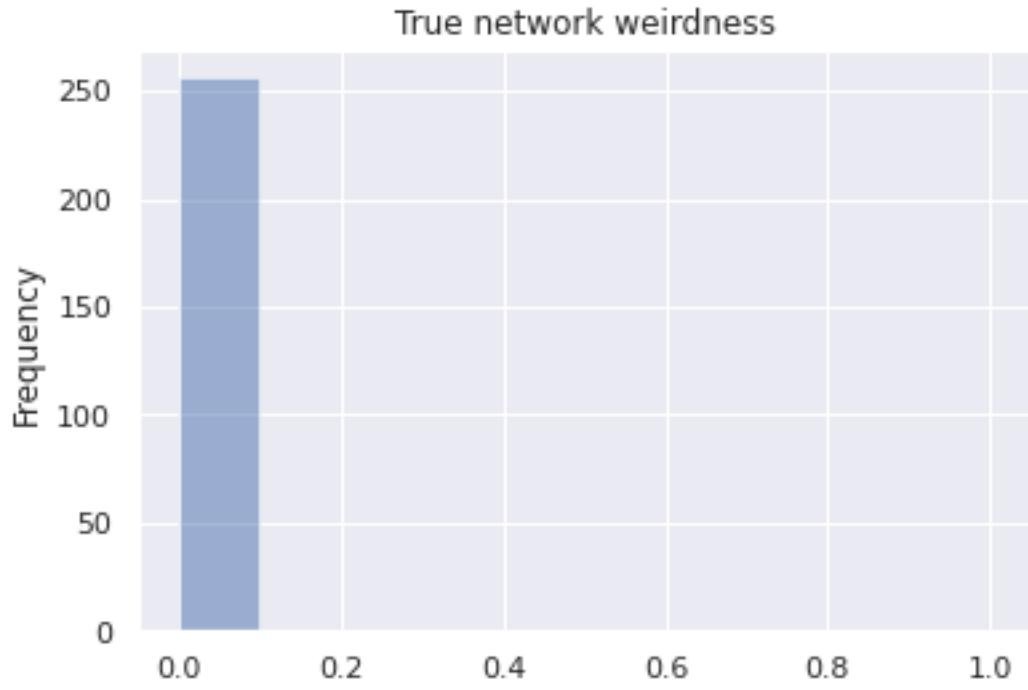
$(bic\_true - bic\_inferred) / bic\_true$   
value >0 means inferred BIC was better



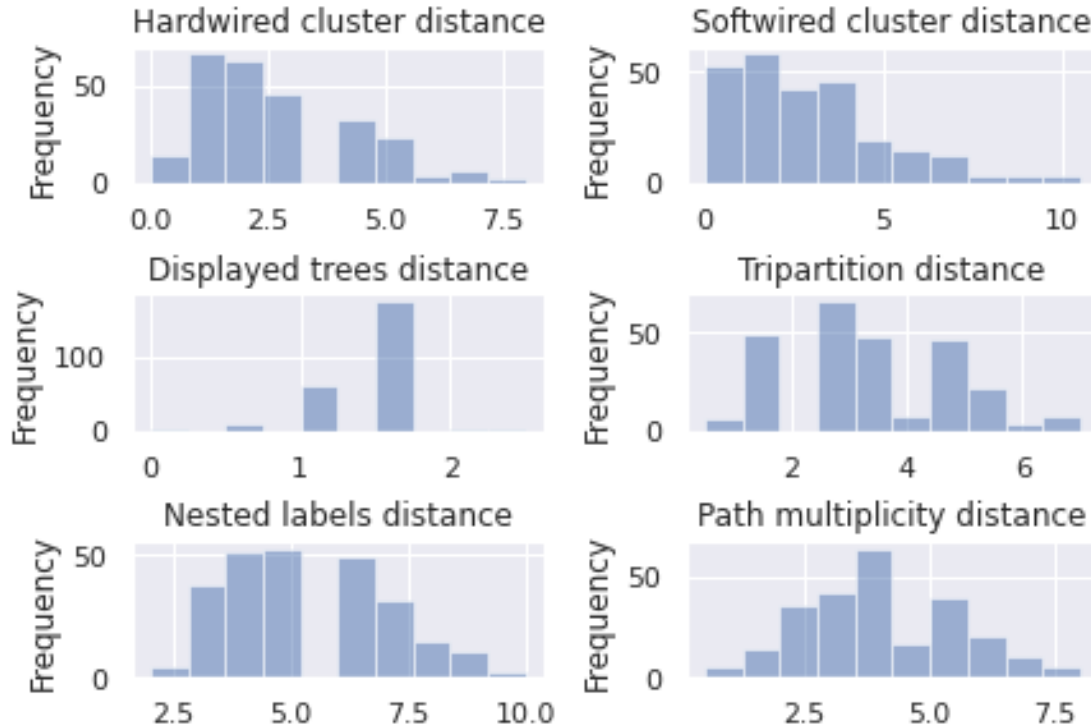
$(logl\_true - logl\_inferred) / logl\_true$   
value <0 means inferred logl was better



<Figure size 432x288 with 0 Axes>



<Figure size 432x288 with 0 Axes>



### 1.2.1 Plots for LikelihoodType.AVERAGE

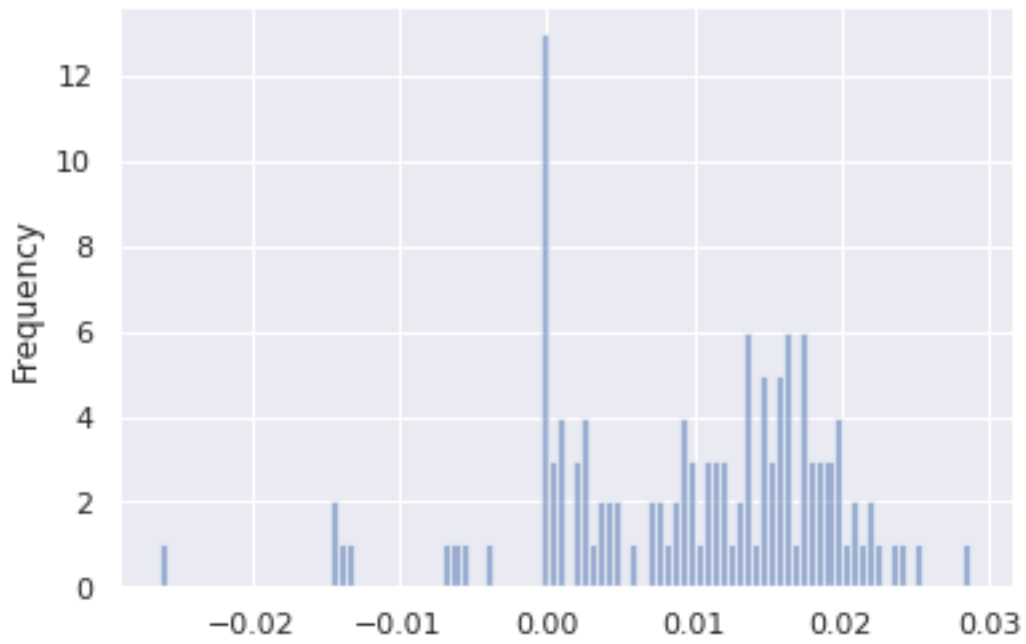
```
[12]: df_raxml_only_msasize_200_average = df_raxml_only_msasize_200.  
      ↪query('likelihood_type == "AVERAGE"')  
      build_stats(df_raxml_only_msasize_200_average)
```

Inferred BIC better or equal: 115  
Inferred BIC worse: 13

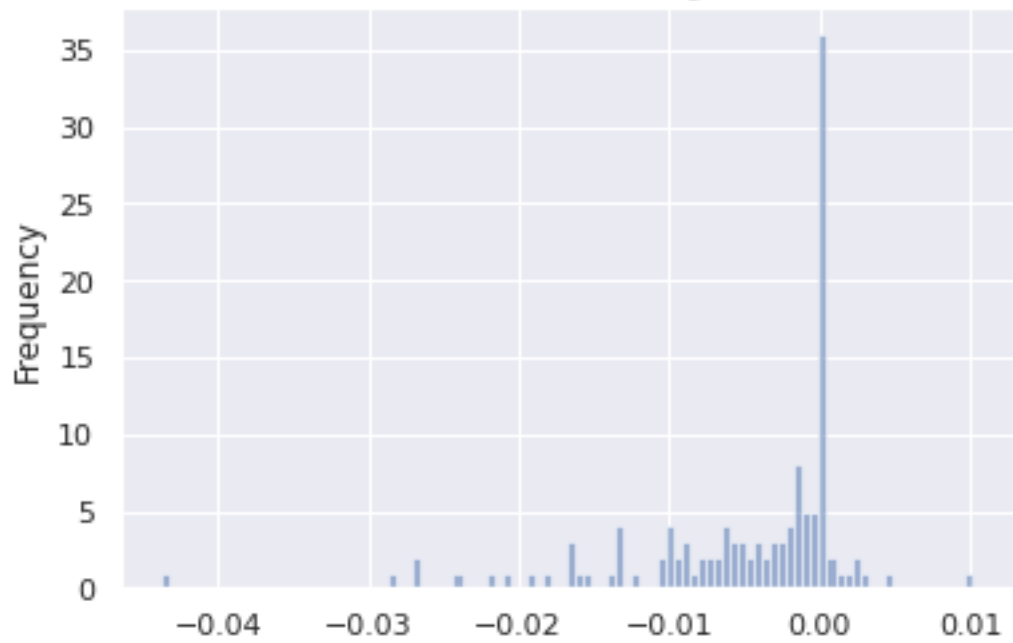
Inferred loglh better or equal: 38  
Inferred loglh worse: 90

Inferred n\_reticulations less: 104  
Inferred n\_reticulations equal: 24  
Inferred n\_reticulations more: 0

$(bic\_true - bic\_inferred) / bic\_true$   
value >0 means inferred BIC was better

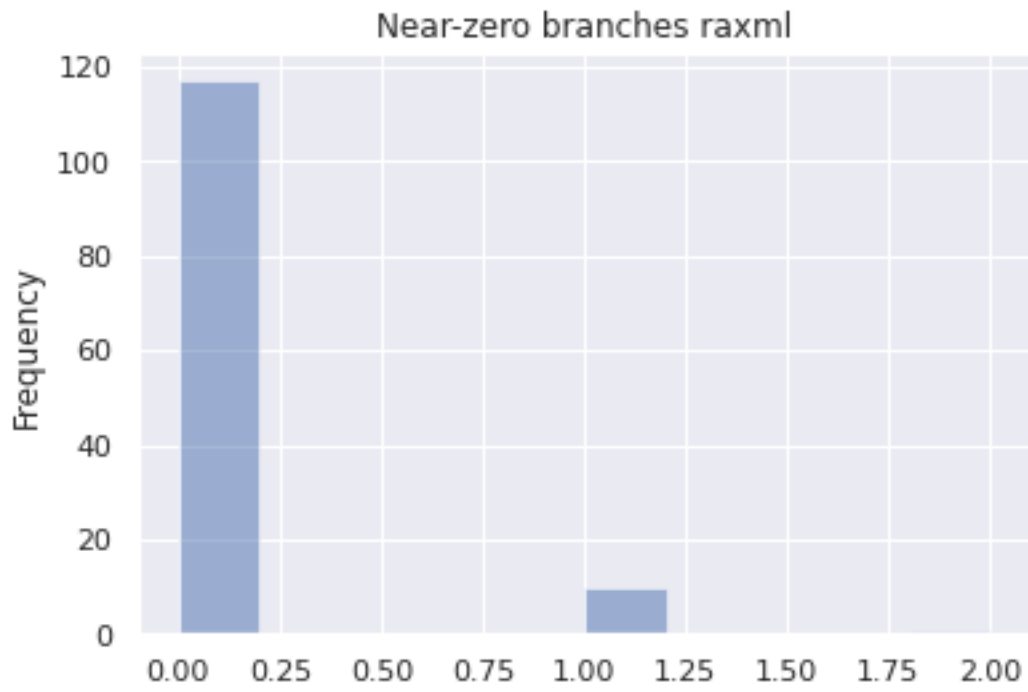
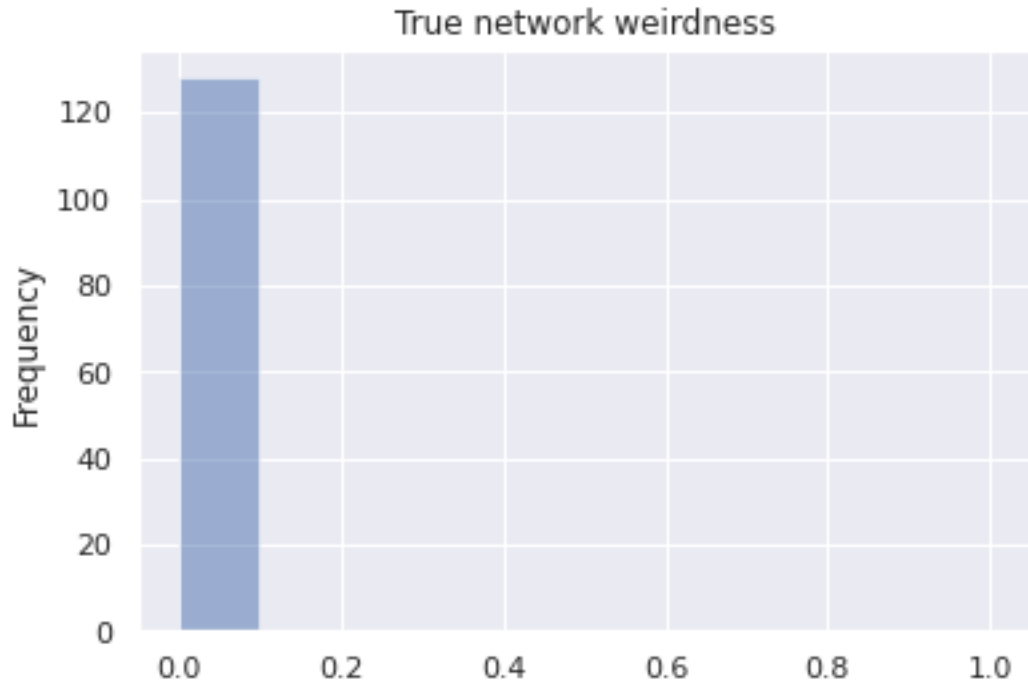


$(logl\_true - logl\_inferred) / logl\_true$   
value <0 means inferred logl was better

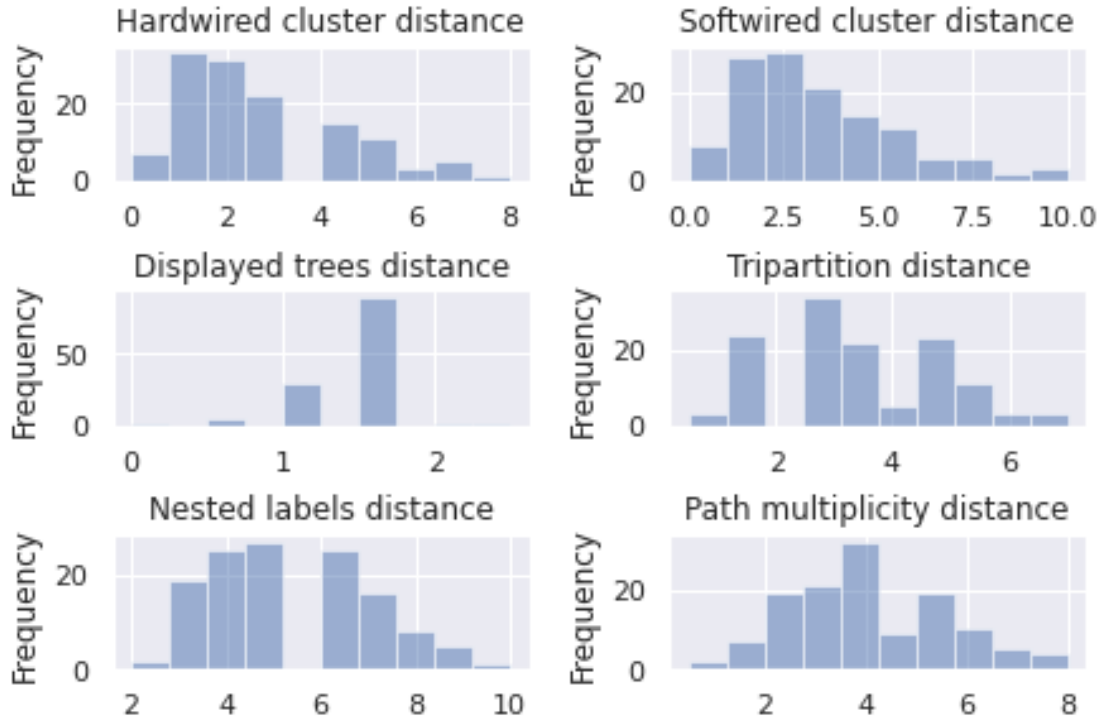




<Figure size 432x288 with 0 Axes>



<Figure size 432x288 with 0 Axes>



### 1.2.2 Plots for LikelihoodType.BEST

```
[13]: df_raxml_only_msasize_200_best = df_raxml_only_msasize_200.  
      ↪query('likelihood_type == "BEST"')  
      build_stats(df_raxml_only_msasize_200_best)
```

Inferred BIC better or equal: 115

Inferred BIC worse: 13

Inferred loglh better or equal: 41

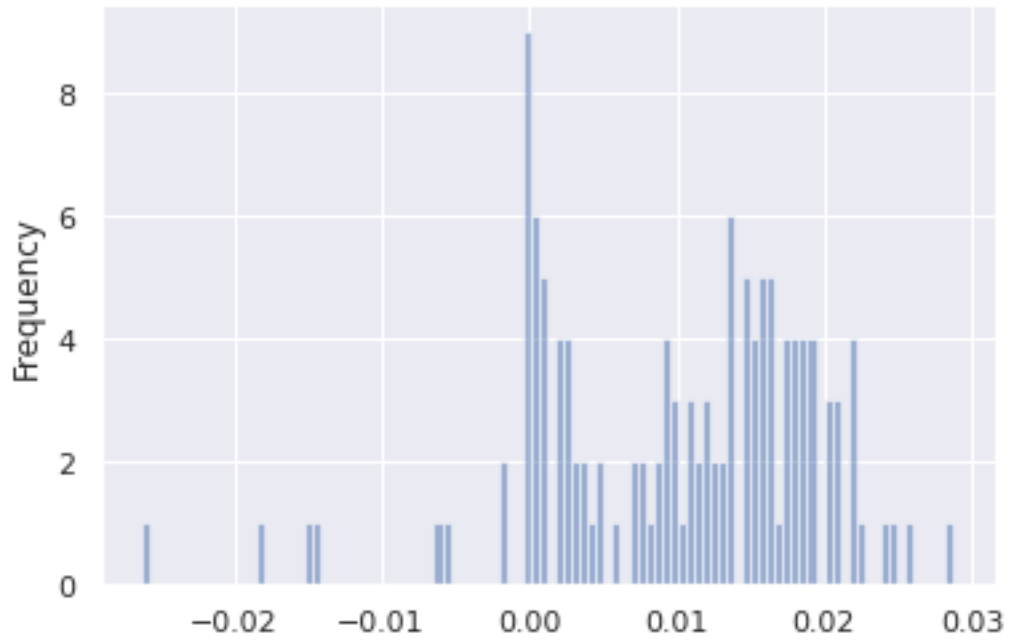
Inferred loglh worse: 87

Inferred n\_reticulations less: 106

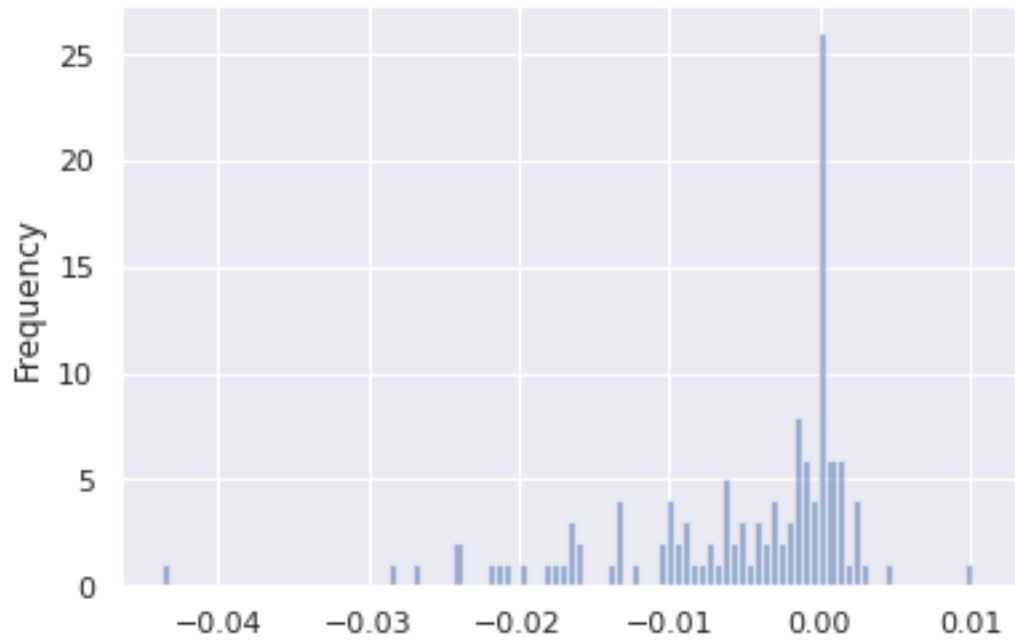
Inferred n\_reticulations equal: 22

Inferred n\_reticulations more: 0

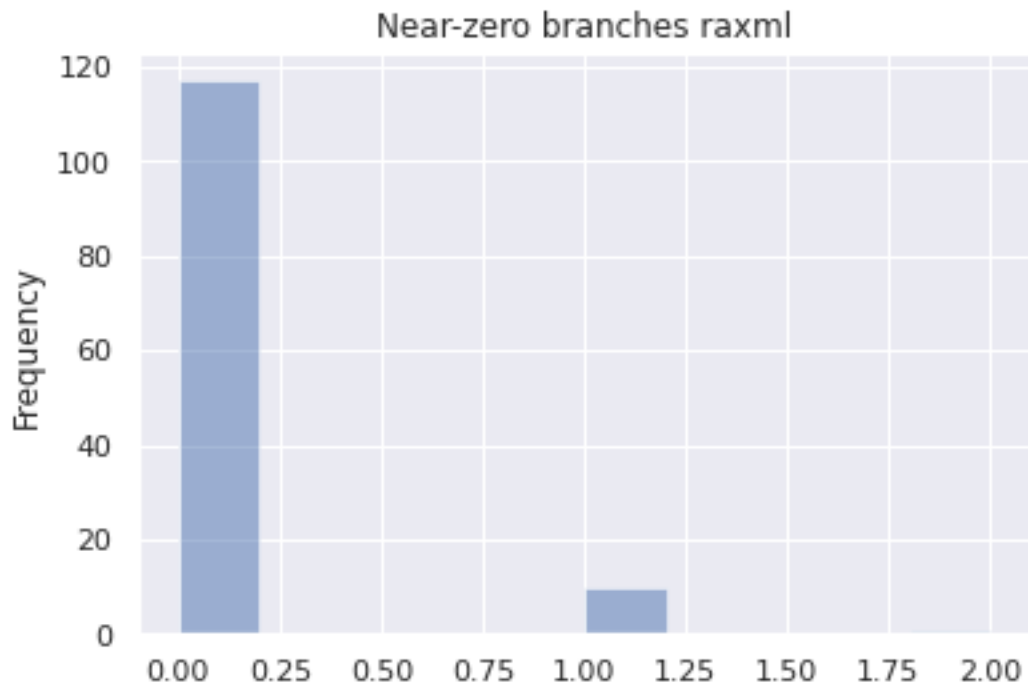
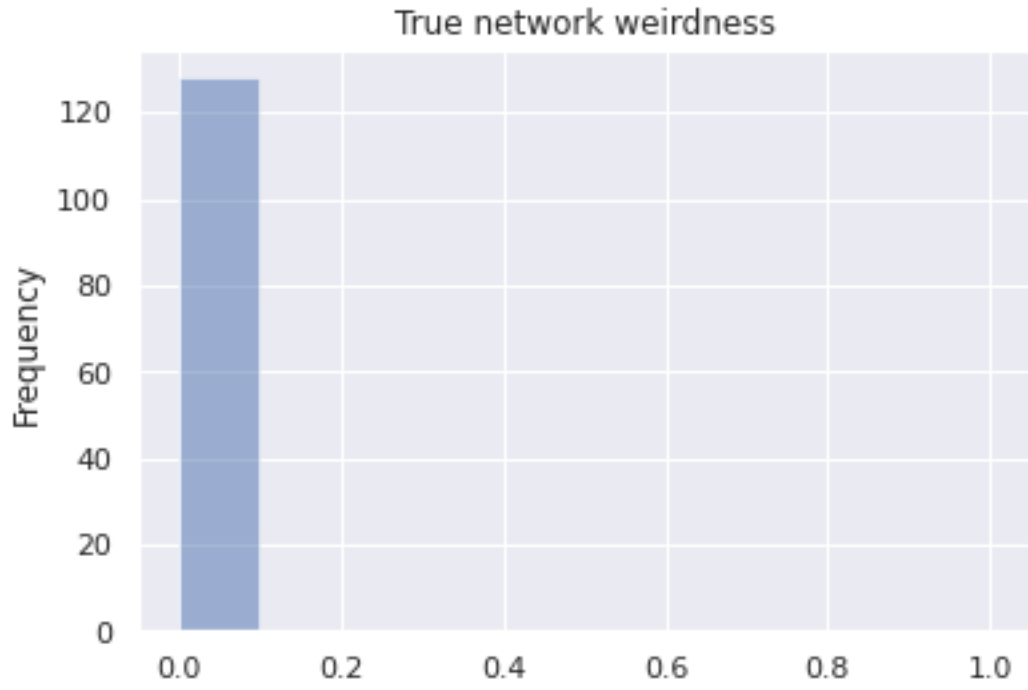
$(bic\_true - bic\_inferred) / bic\_true$   
value >0 means inferred BIC was better



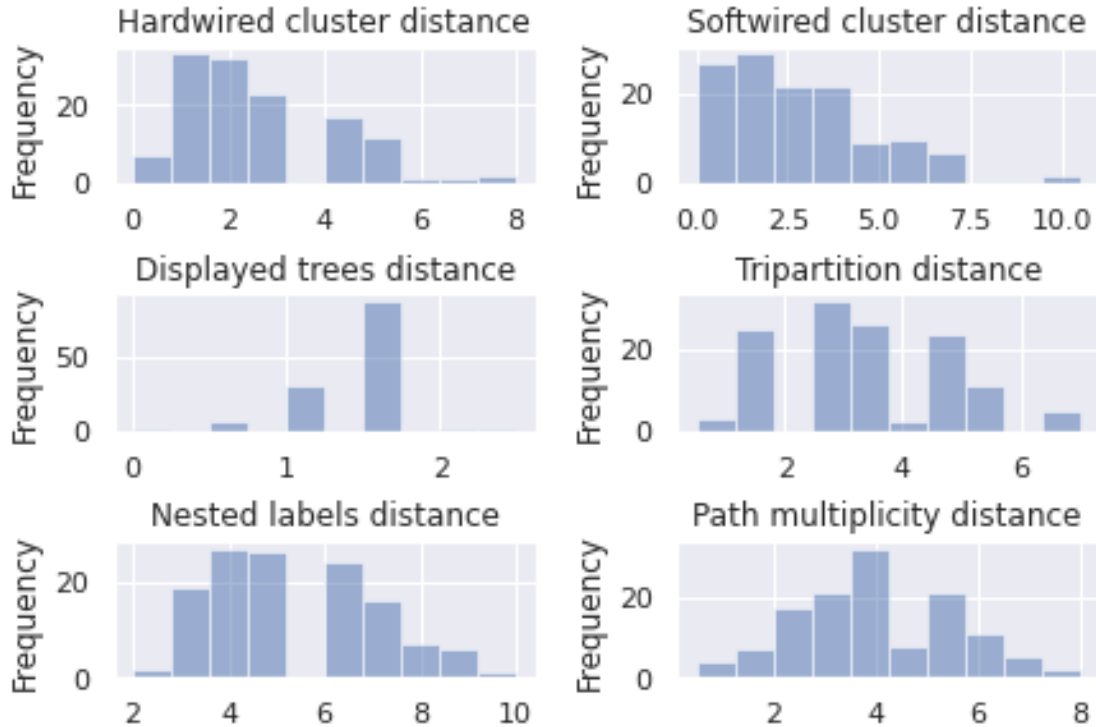
$(logl\_true - logl\_inferred) / logl\_true$   
value <0 means inferred logl was better



<Figure size 432x288 with 0 Axes>



<Figure size 432x288 with 0 Axes>



## 2 Plots for starting with 5 random, 5 parsimony trees

```
[14]: df_random = df.query('start_type == "RANDOM"')
      build_stats(df_random)
```

Inferred BIC better or equal: 505

Inferred BIC worse: 7

Inferred loglh better or equal: 201

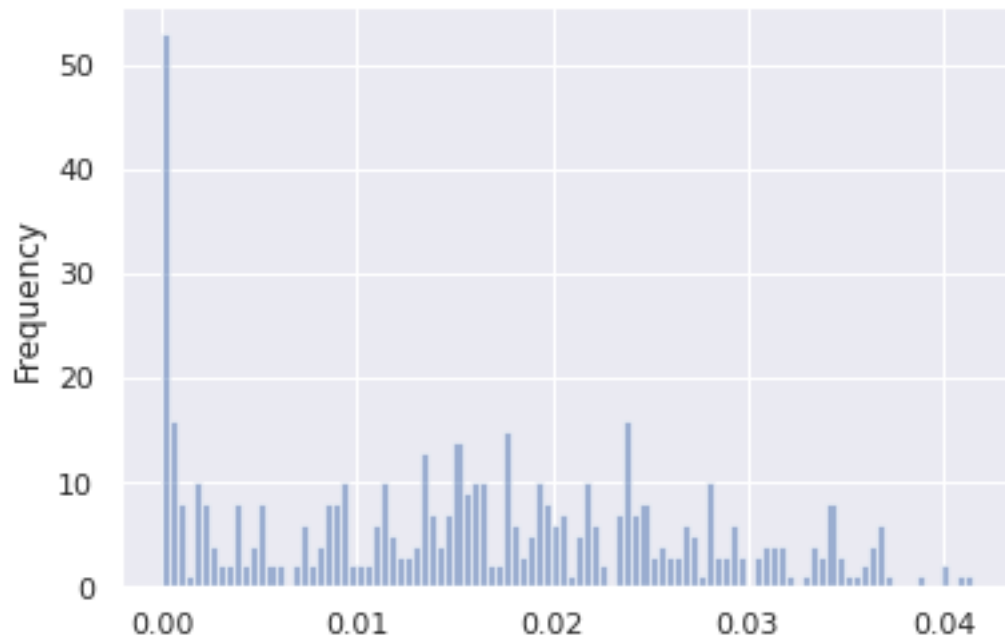
Inferred loglh worse: 311

Inferred n\_reticulations less: 428

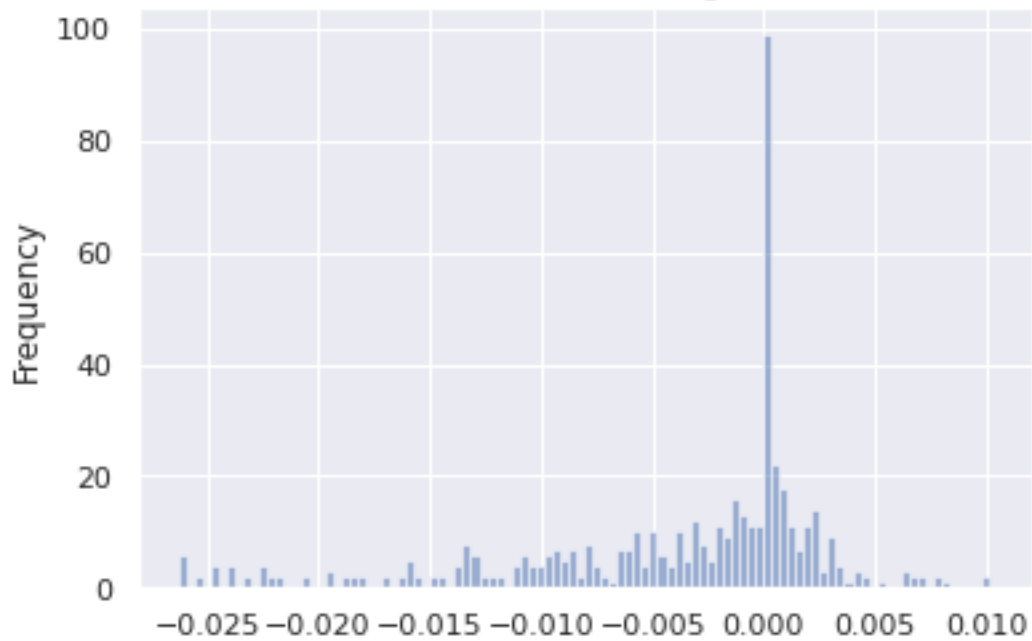
Inferred n\_reticulations equal: 84

Inferred n\_reticulations more: 0

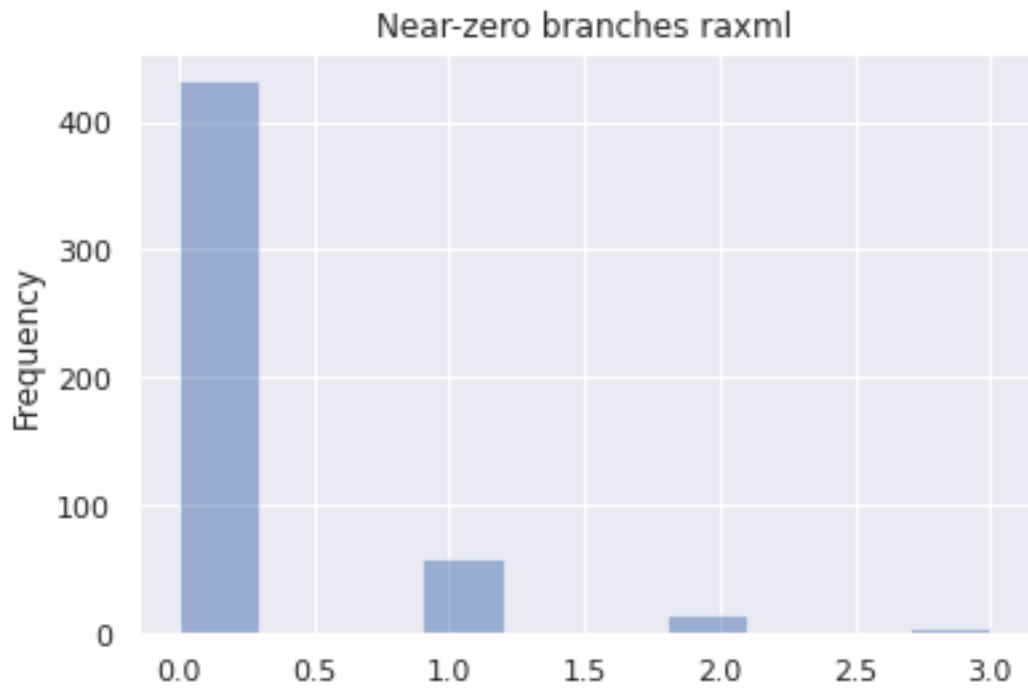
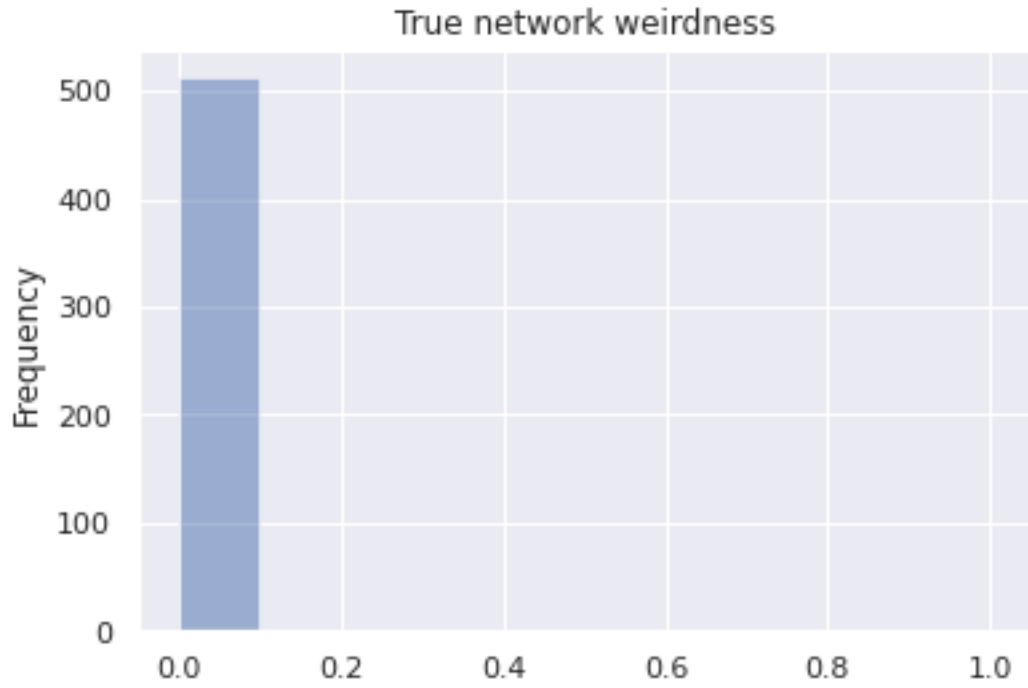
$(bic\_true - bic\_inferred) / bic\_true$   
value >0 means inferred BIC was better



$(logl\_true - logl\_inferred) / logl\_true$   
value <0 means inferred logl was better



<Figure size 432x288 with 0 Axes>



<Figure size 432x288 with 0 Axes>



## 2.1 Plots for MSA\_size ~ 100\*n\_trees

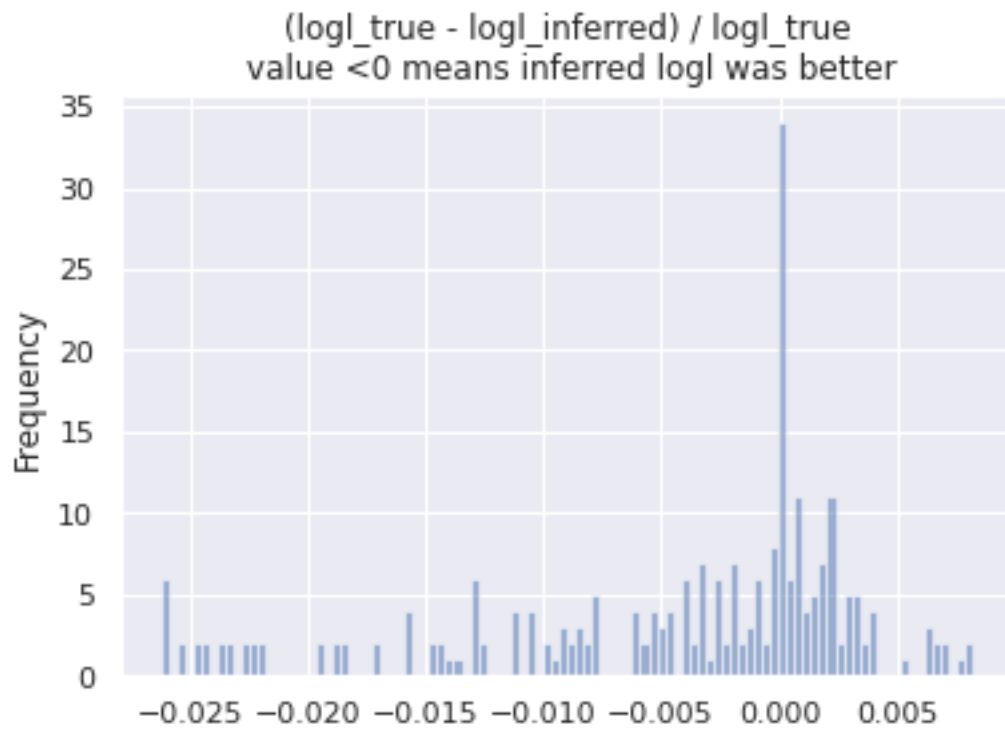
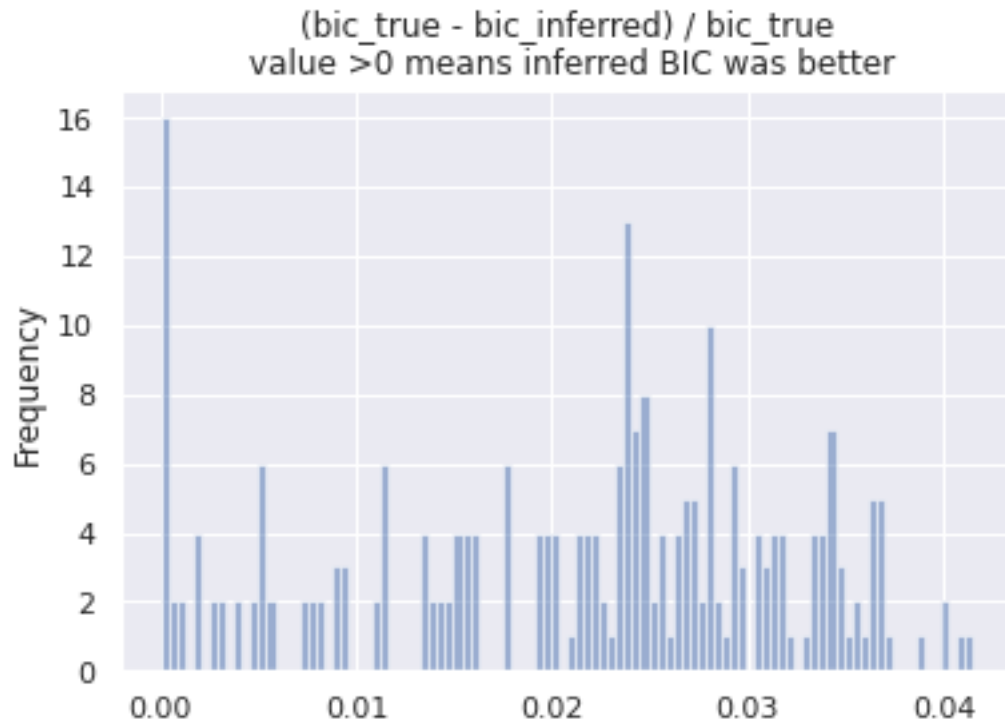
```
[15]: df_random_msasize_100 = df_random.query('msa_size == 100')
      build_stats(df_random_msasize_100)
```

Inferred BIC better or equal: 250  
Inferred BIC worse: 0

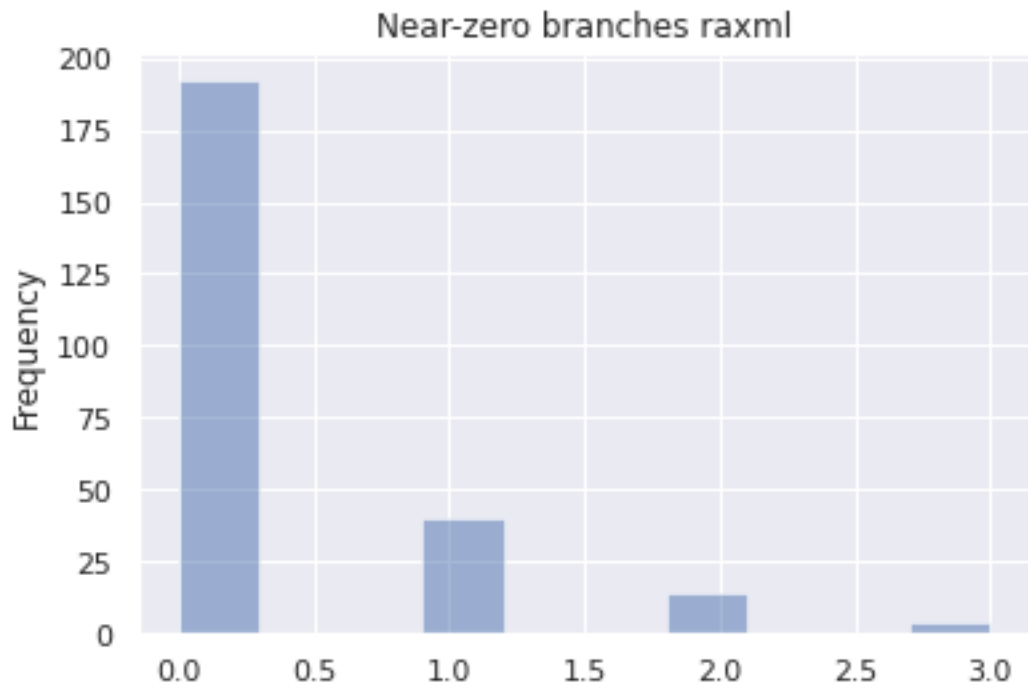
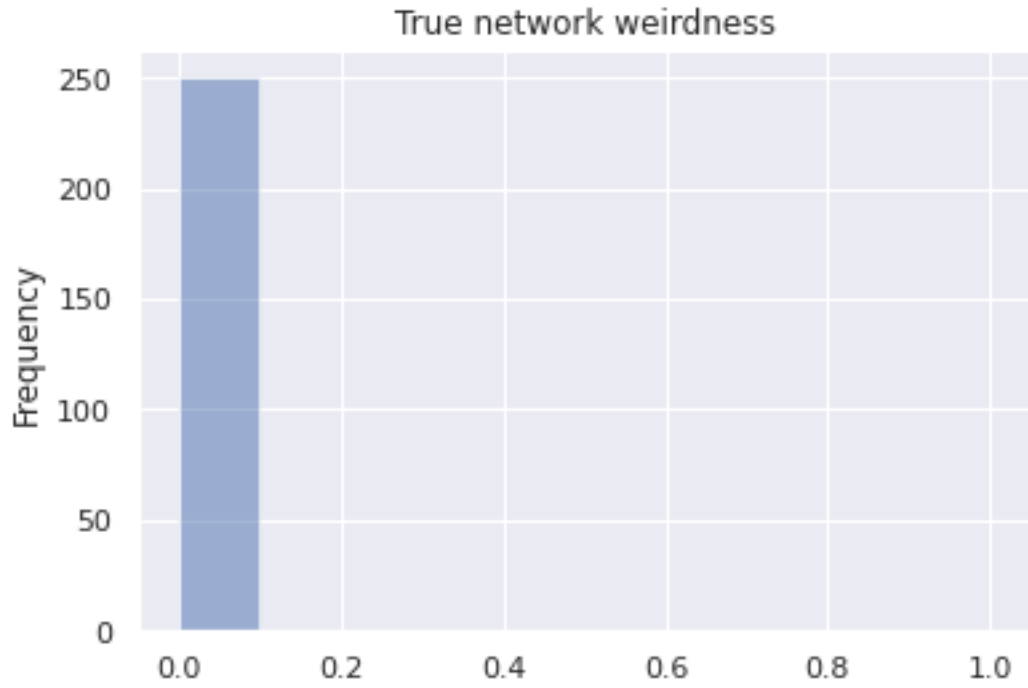
Inferred loglh better or equal: 105  
Inferred loglh worse: 145

Inferred n\_reticulations less: 222  
Inferred n\_reticulations equal: 28  
Inferred n\_reticulations more: 0

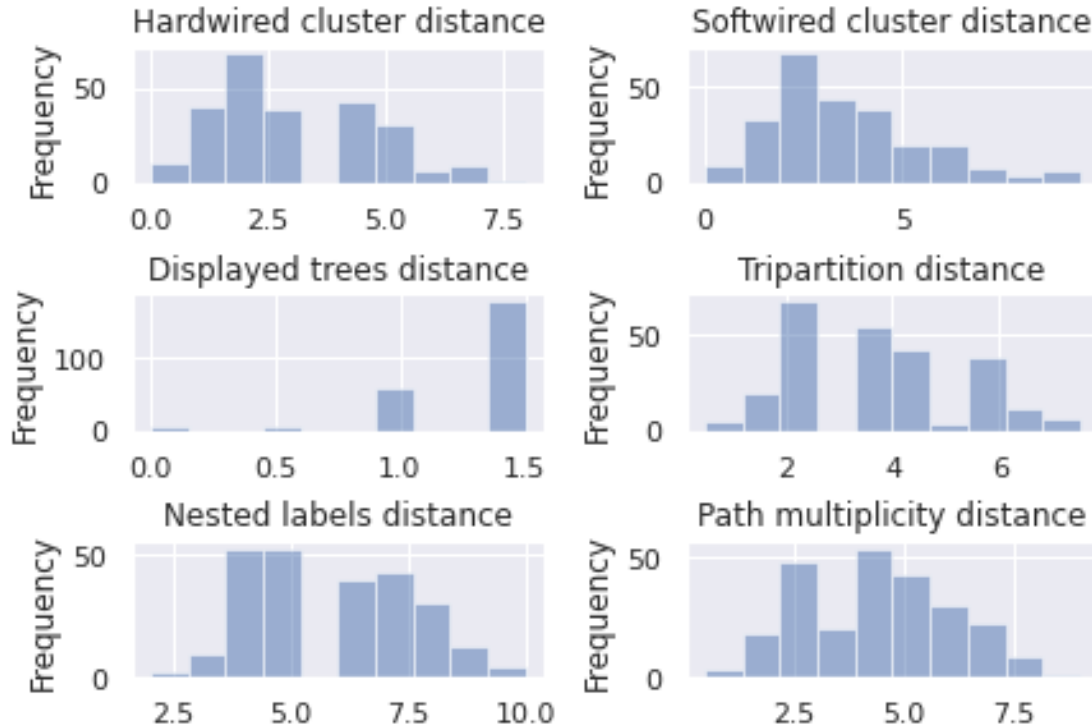




<Figure size 432x288 with 0 Axes>



<Figure size 432x288 with 0 Axes>



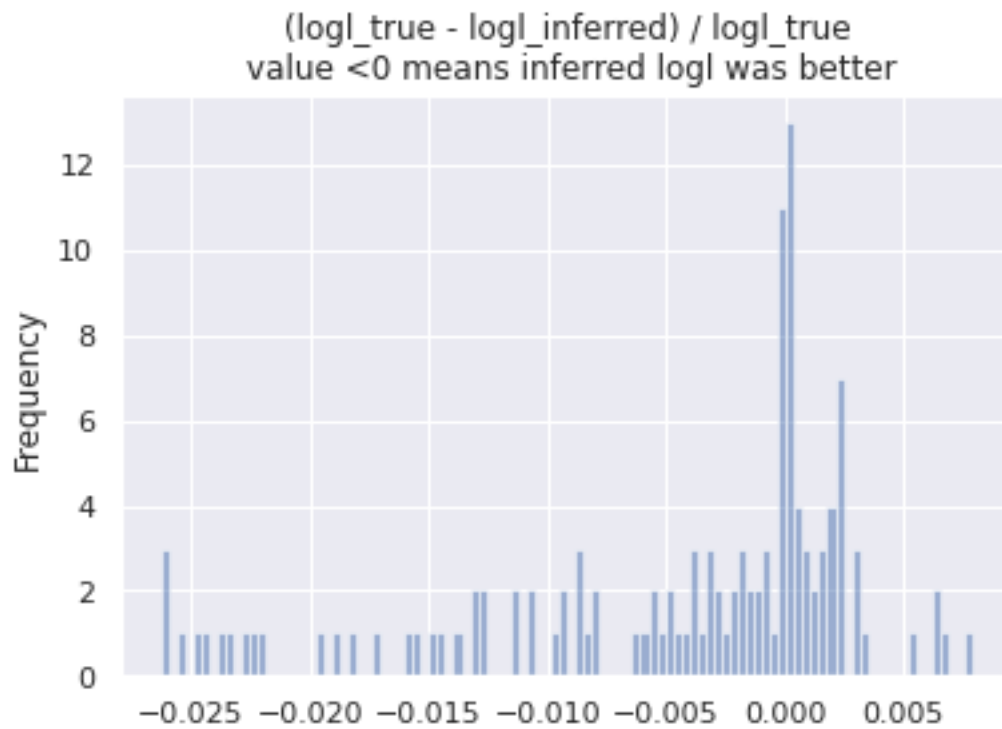
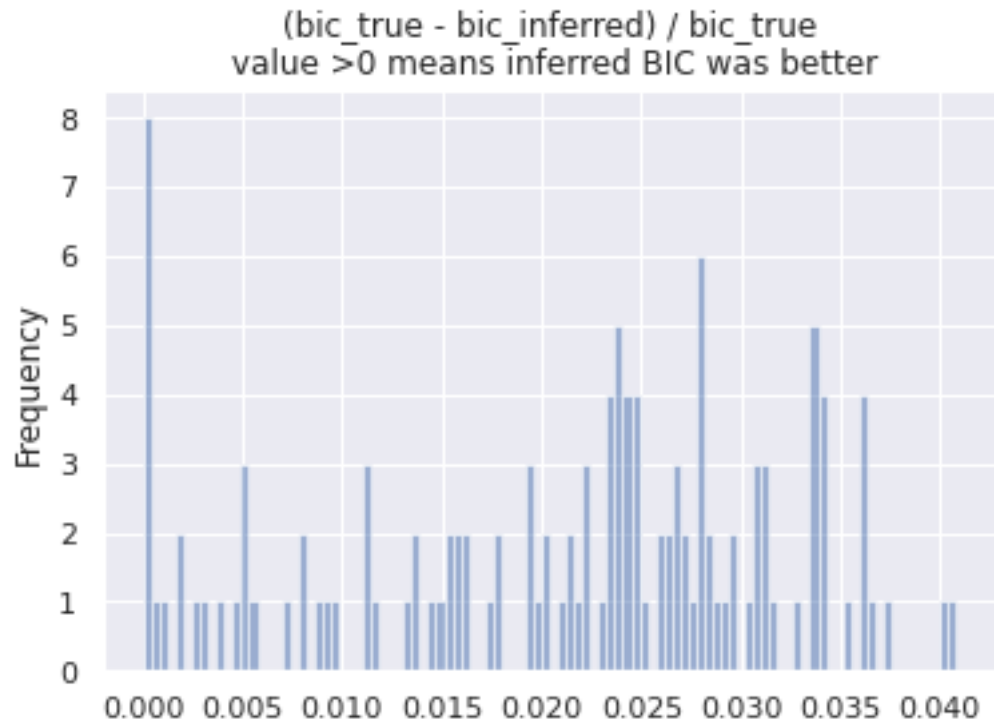
### 2.1.1 Plots for LikelihoodType.AVERAGE

```
[16]: df_random_msasize_100_average = df_random_msasize_100.query('likelihood_type ==  
↳ "AVERAGE"')  
build_stats(df_random_msasize_100_average)
```

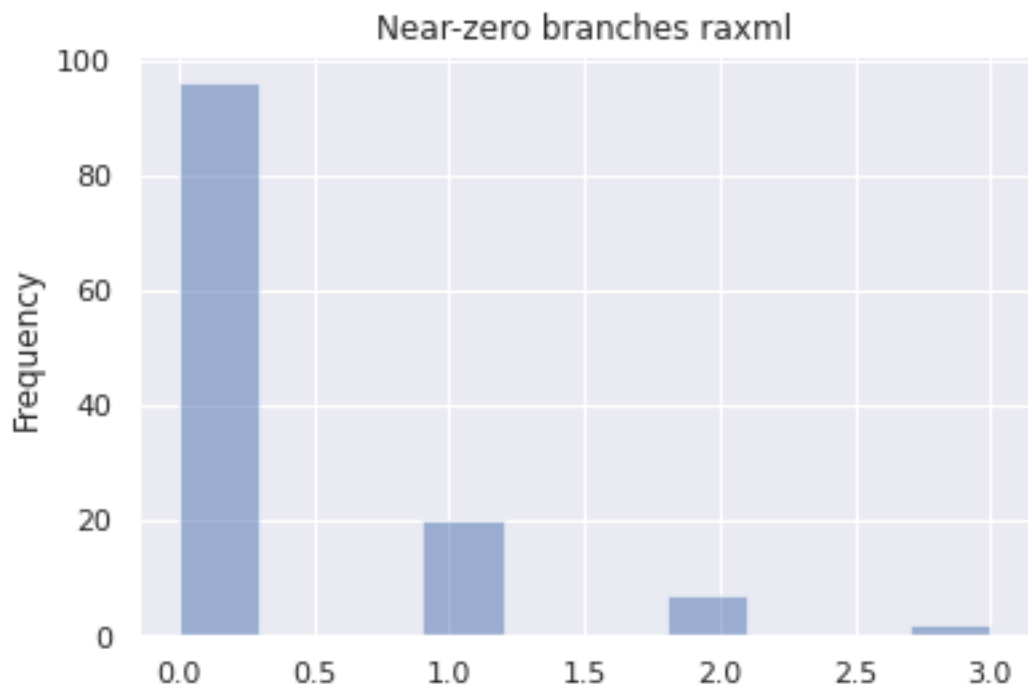
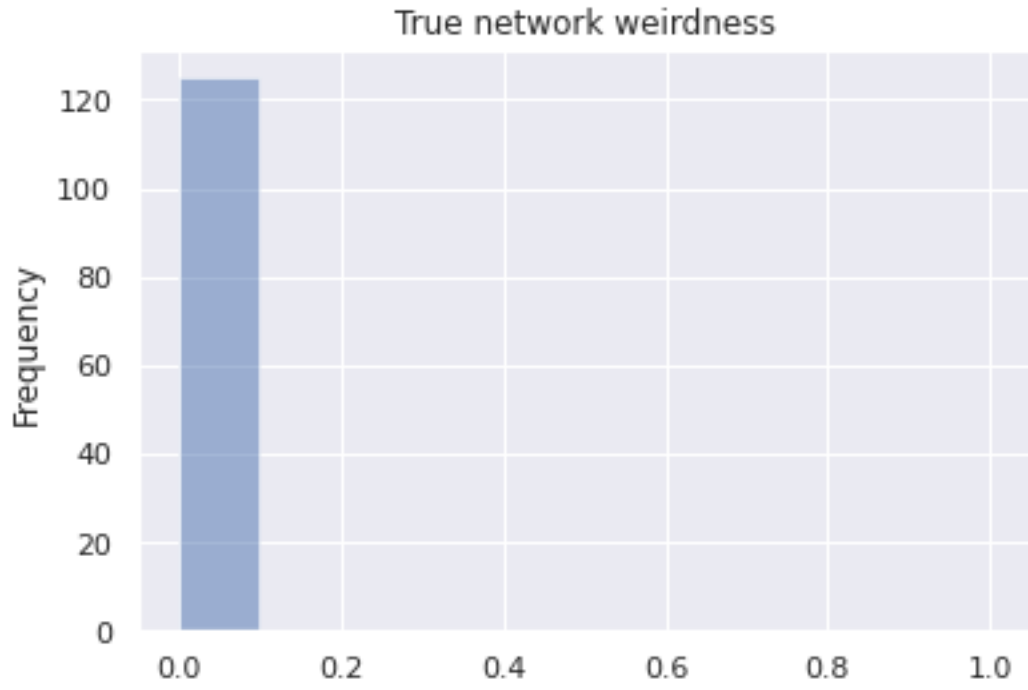
Inferred BIC better or equal: 125  
Inferred BIC worse: 0

Inferred loglh better or equal: 51  
Inferred loglh worse: 74

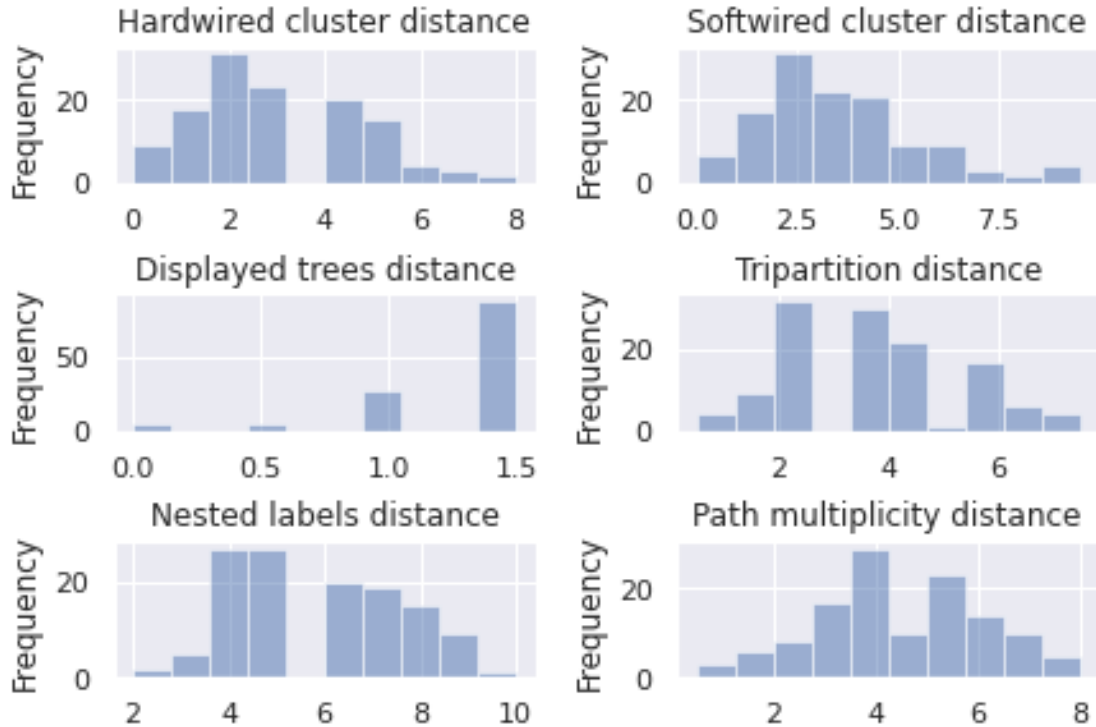
Inferred n\_reticulations less: 111  
Inferred n\_reticulations equal: 14  
Inferred n\_reticulations more: 0



<Figure size 432x288 with 0 Axes>



<Figure size 432x288 with 0 Axes>



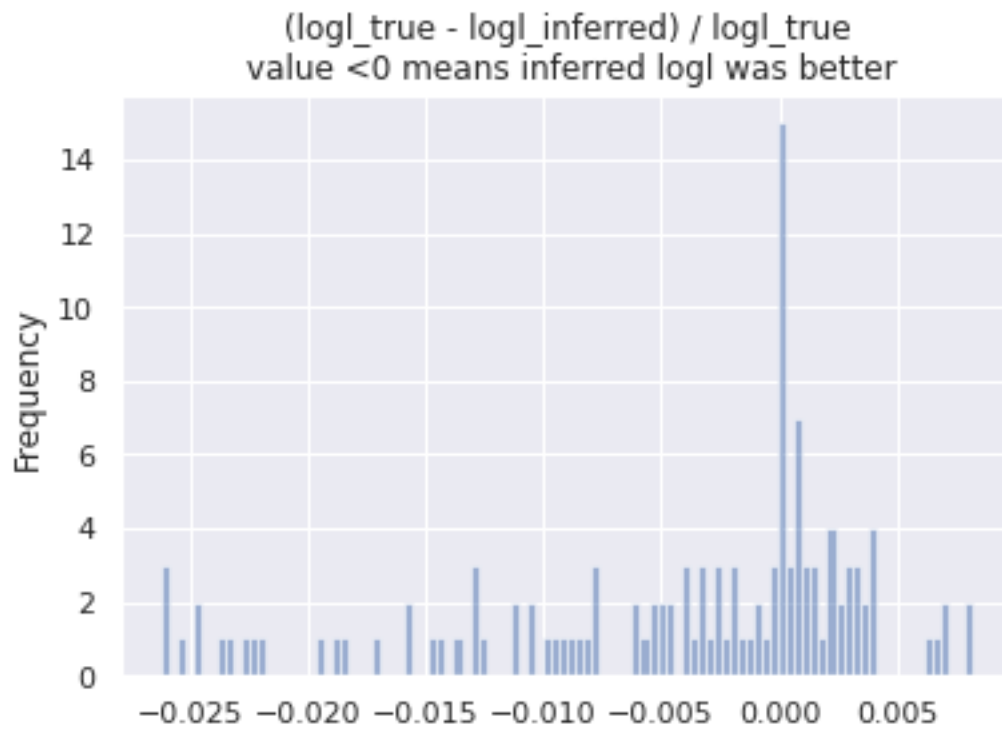
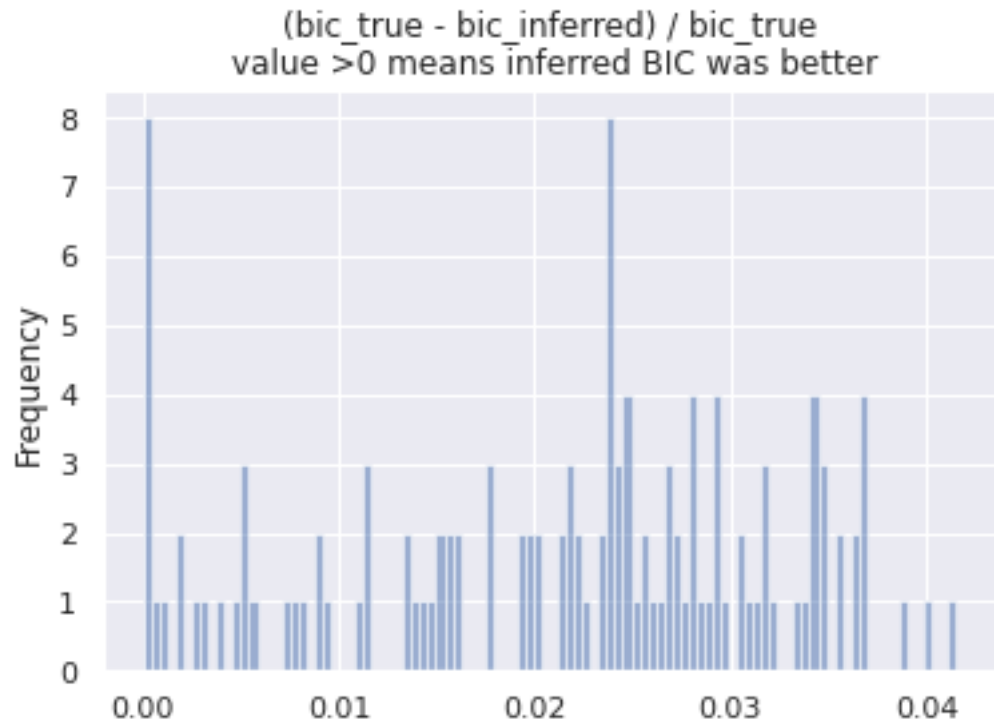
### 2.1.2 Plots for LikelihoodType.BEST

```
[17]: df_random_msasize_100_best = df_random_msasize_100.query('likelihood_type ==  
↳"BEST"')  
build_stats(df_random_msasize_100_best)
```

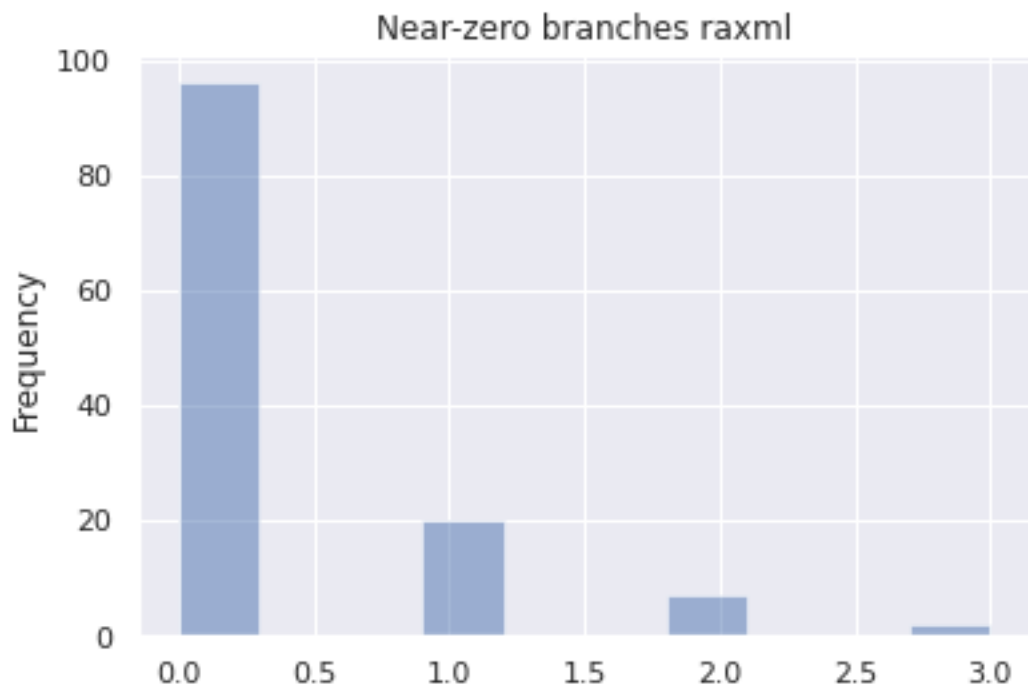
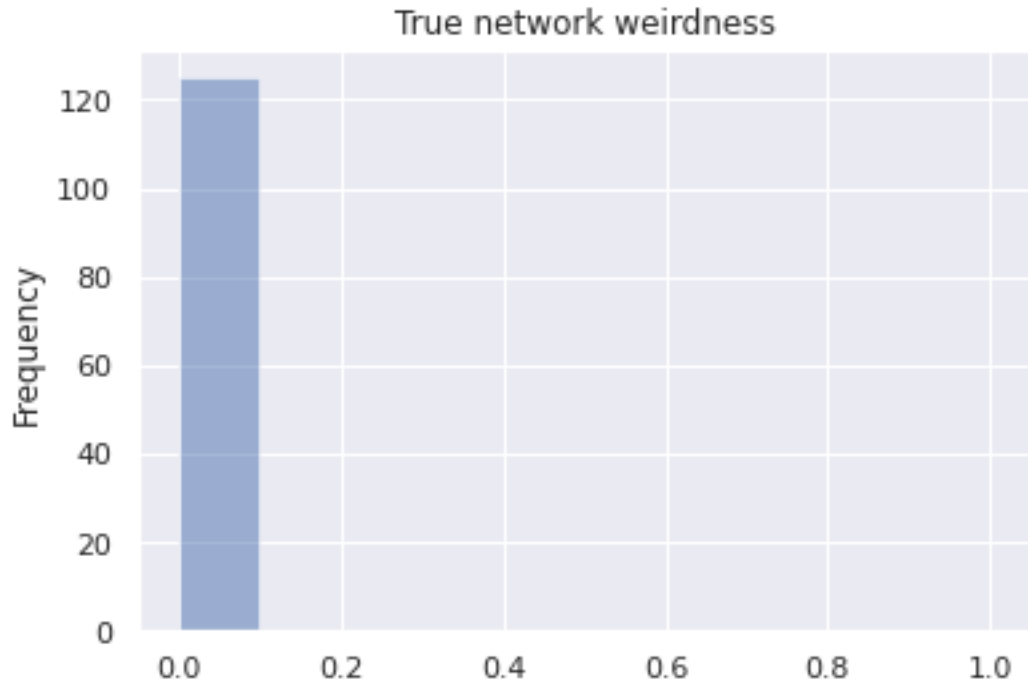
Inferred BIC better or equal: 125  
Inferred BIC worse: 0

Inferred loglh better or equal: 54  
Inferred loglh worse: 71

Inferred n\_reticulations less: 111  
Inferred n\_reticulations equal: 14  
Inferred n\_reticulations more: 0



<Figure size 432x288 with 0 Axes>





<Figure size 432x288 with 0 Axes>



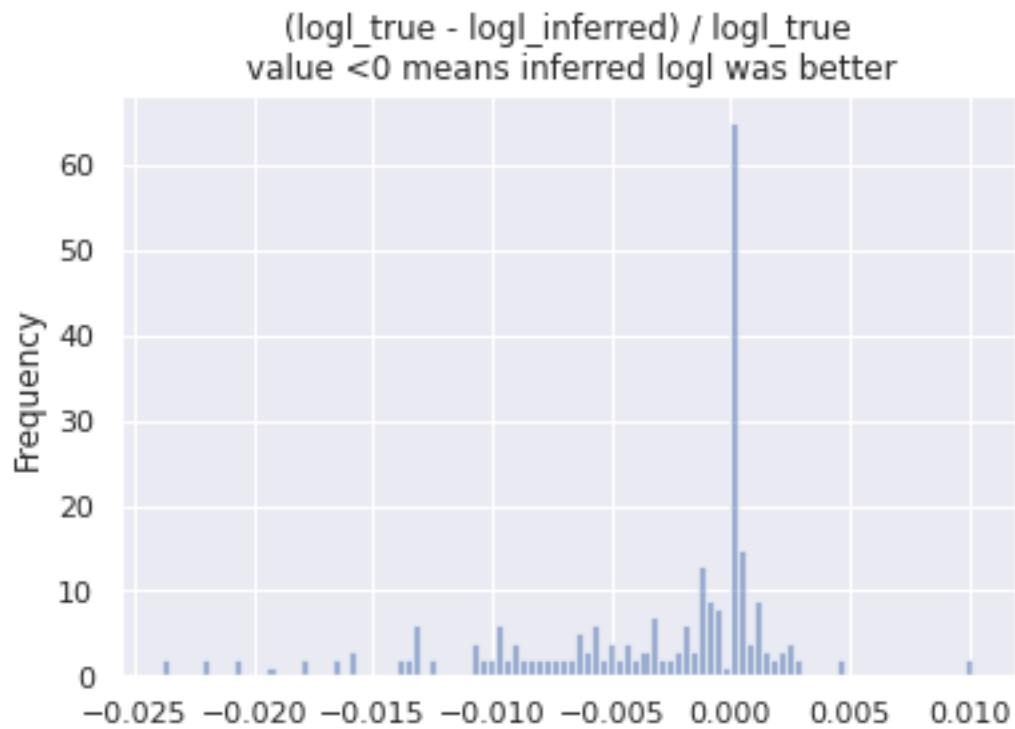
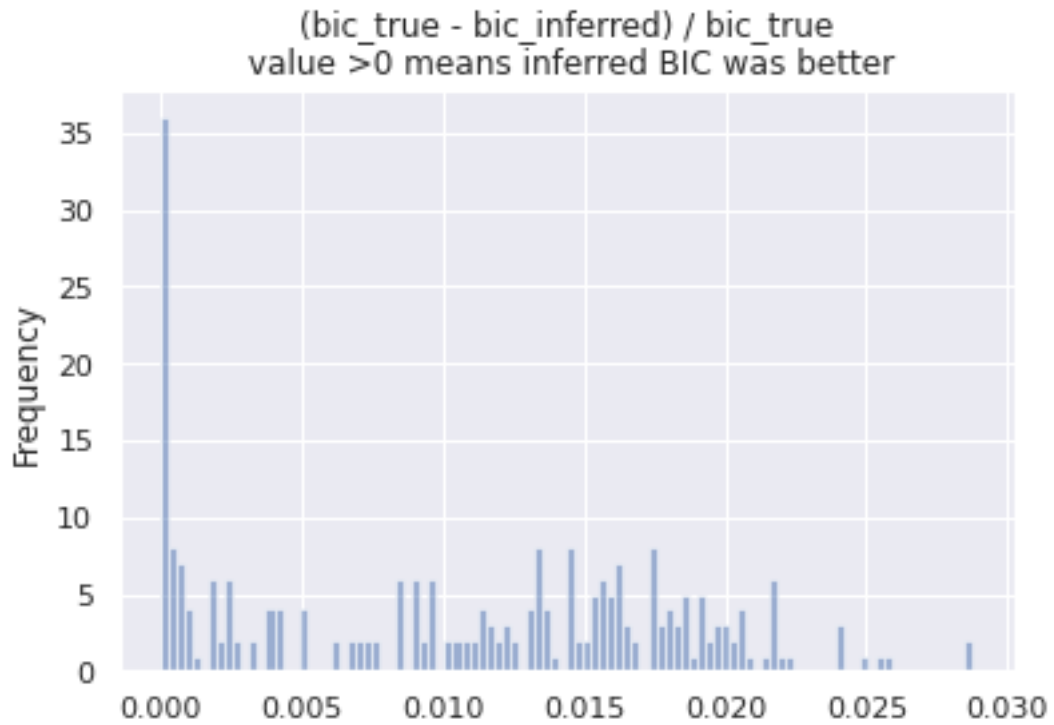
## 2.2 Plots for $MSA\_size \sim 200 * n\_trees$

```
[18]: df_random_msasize_200 = df_random.query('msa_size == 200')
      build_stats(df_random_msasize_200)
```

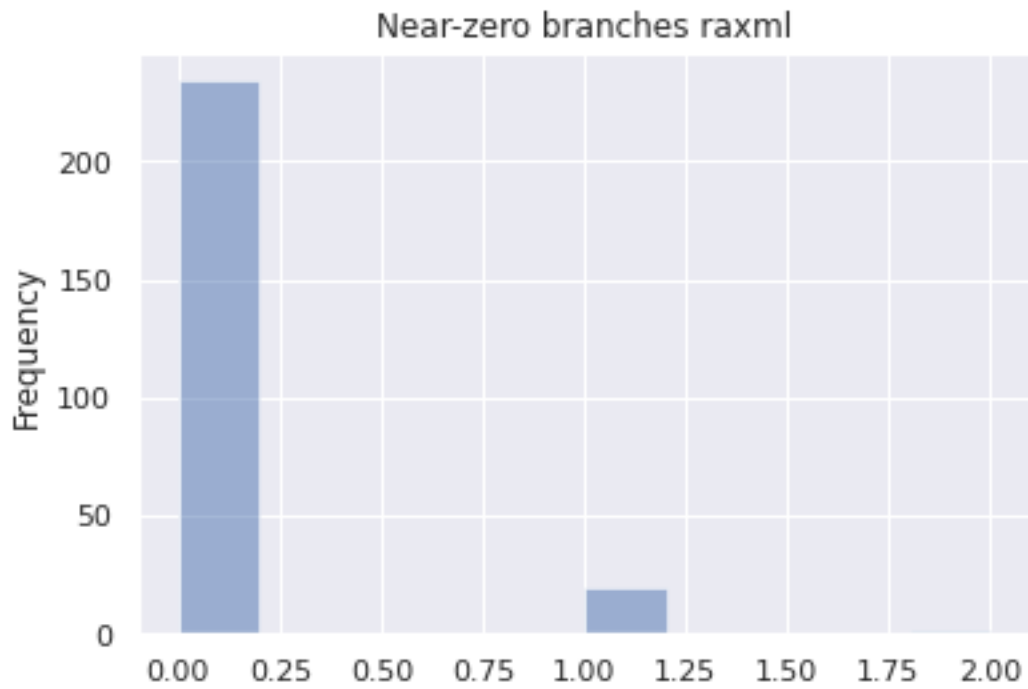
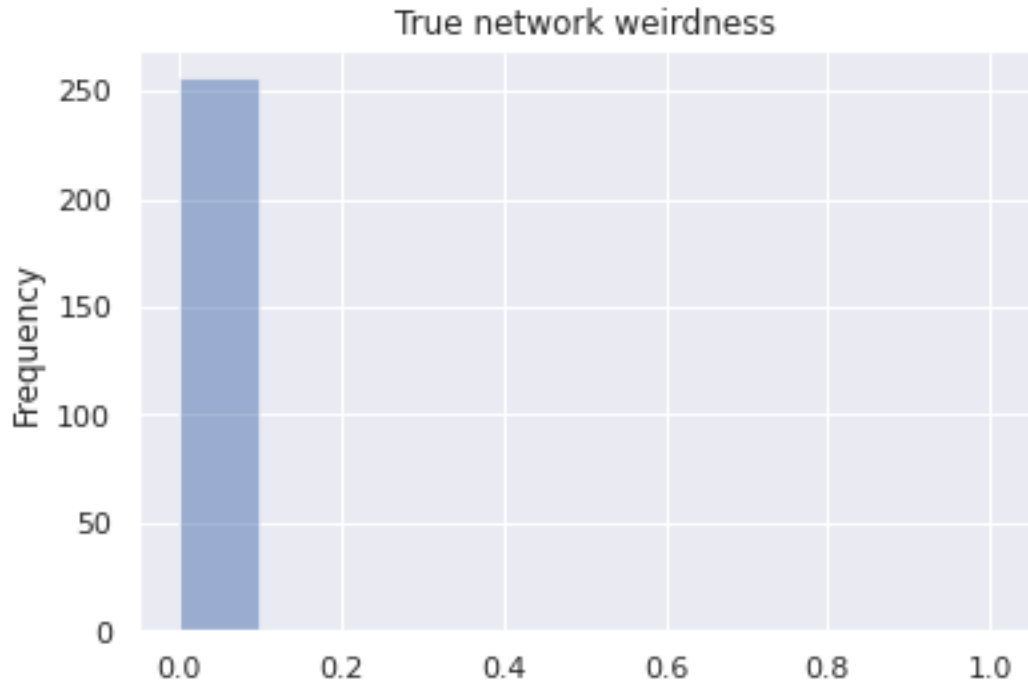
Inferred BIC better or equal: 249  
Inferred BIC worse: 7

Inferred loglh better or equal: 96  
Inferred loglh worse: 160

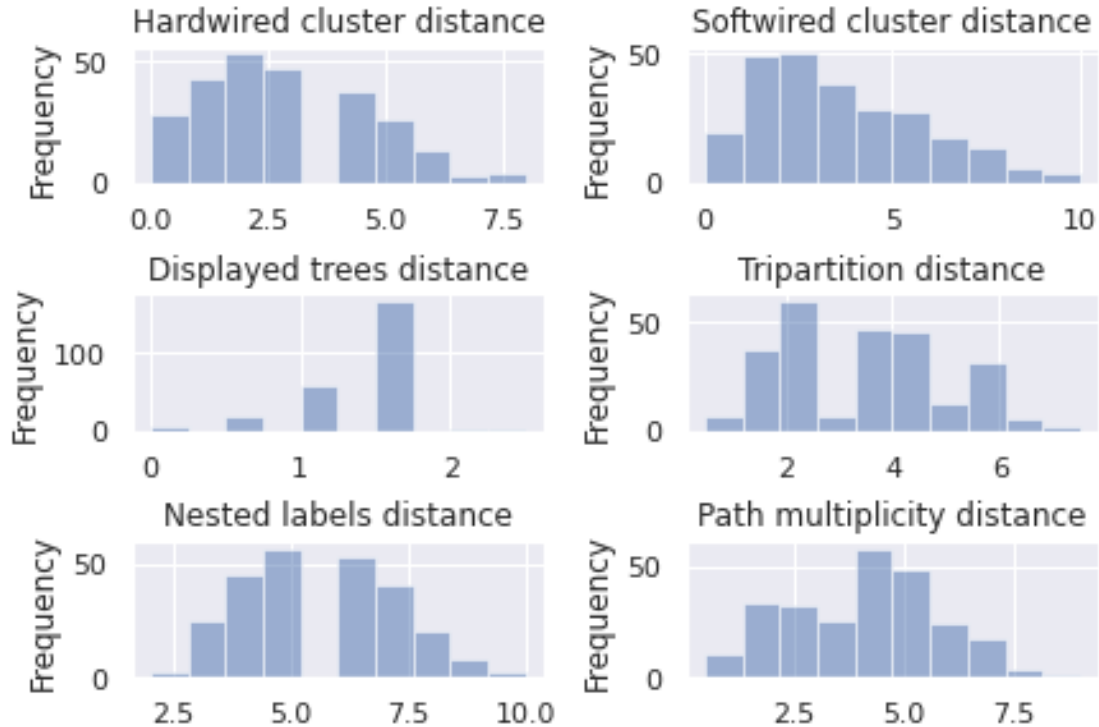
Inferred n\_reticulations less: 200  
Inferred n\_reticulations equal: 56  
Inferred n\_reticulations more: 0



<Figure size 432x288 with 0 Axes>



<Figure size 432x288 with 0 Axes>



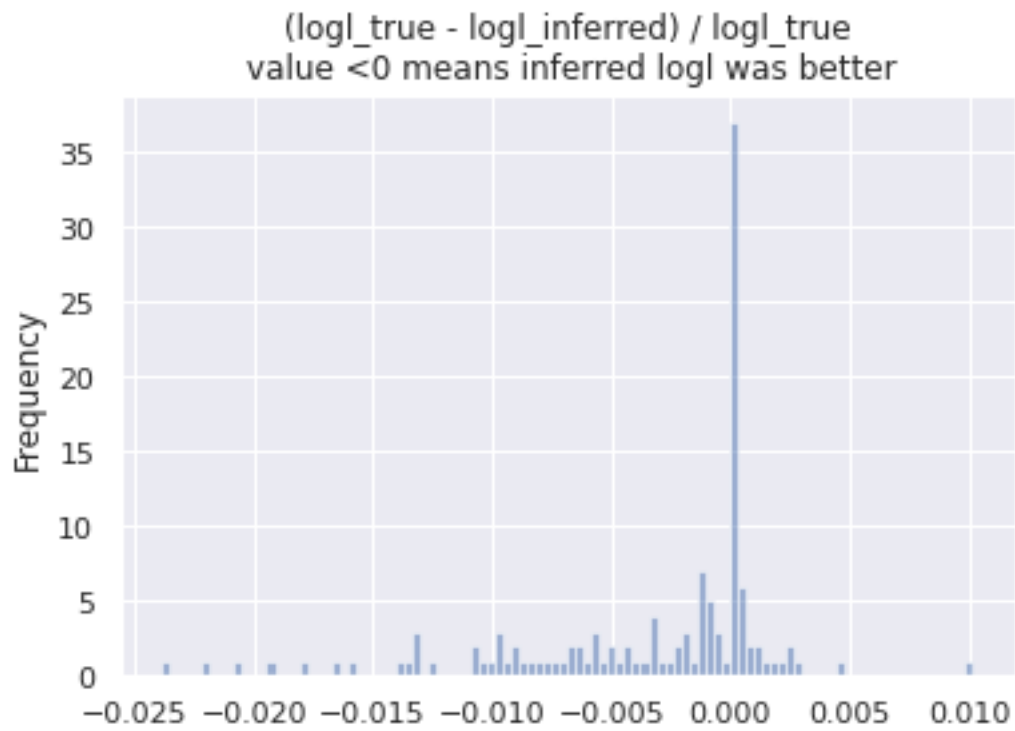
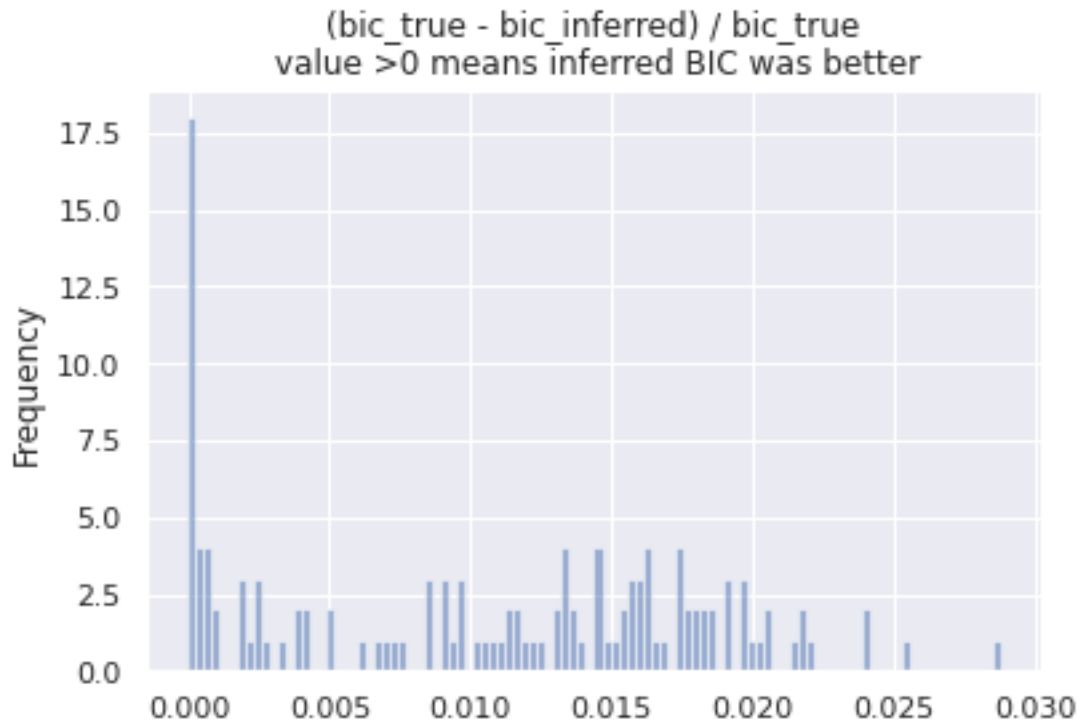
### 2.2.1 Plots for LikelihoodType.AVERAGE

```
[19]: df_random_msasize_200_average = df_random_msasize_200.query('likelihood_type ==  
↳ "AVERAGE"')  
build_stats(df_random_msasize_200_average)
```

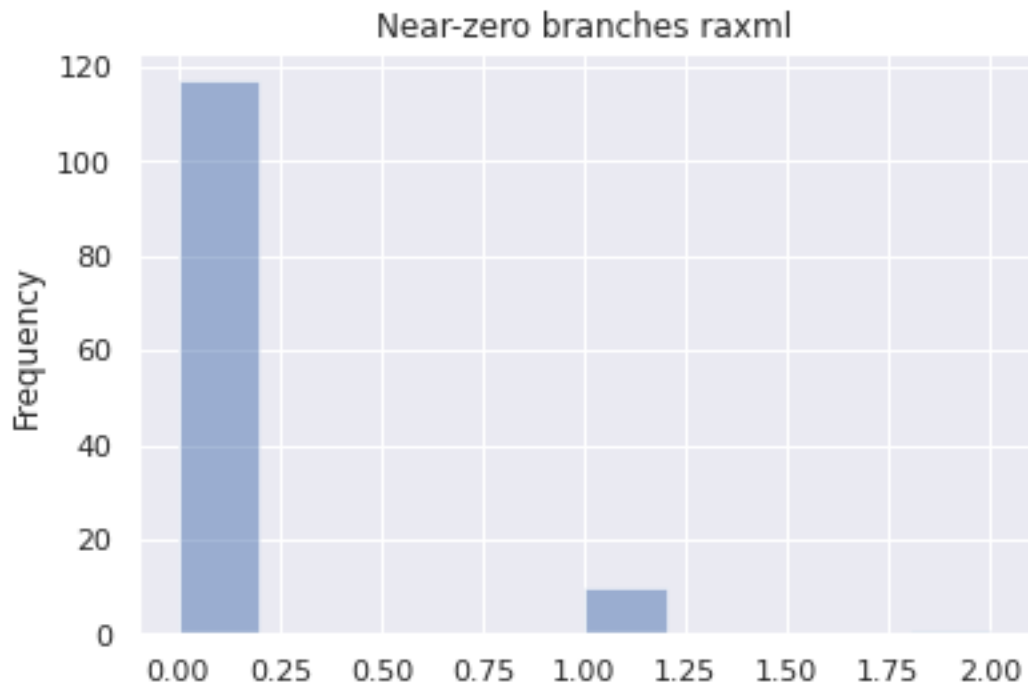
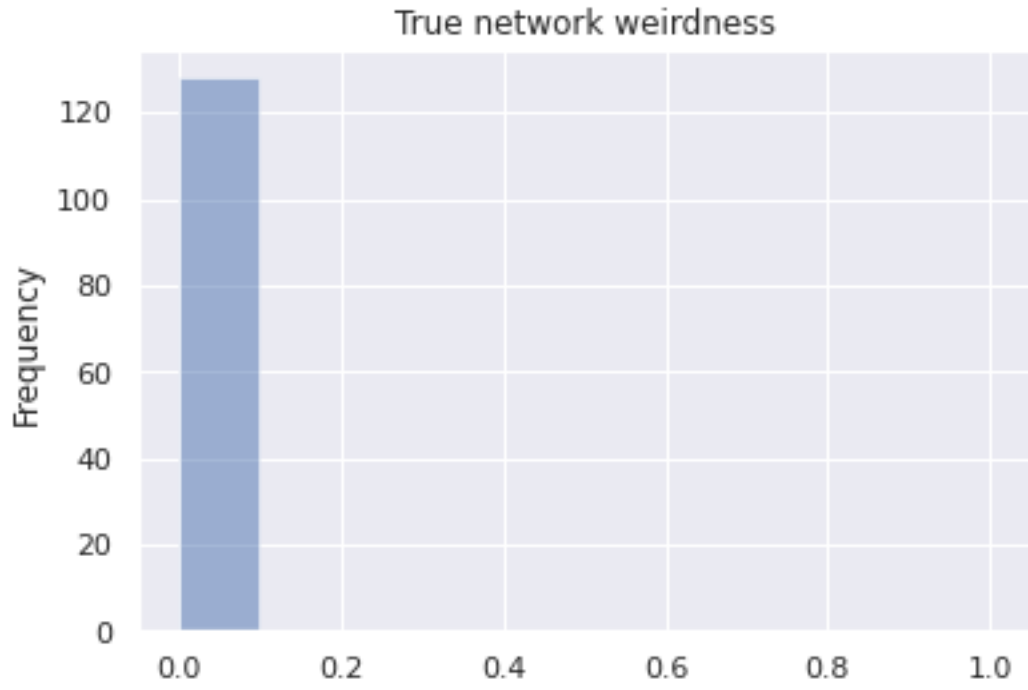
Inferred BIC better or equal: 125  
Inferred BIC worse: 3

Inferred loglh better or equal: 46  
Inferred loglh worse: 82

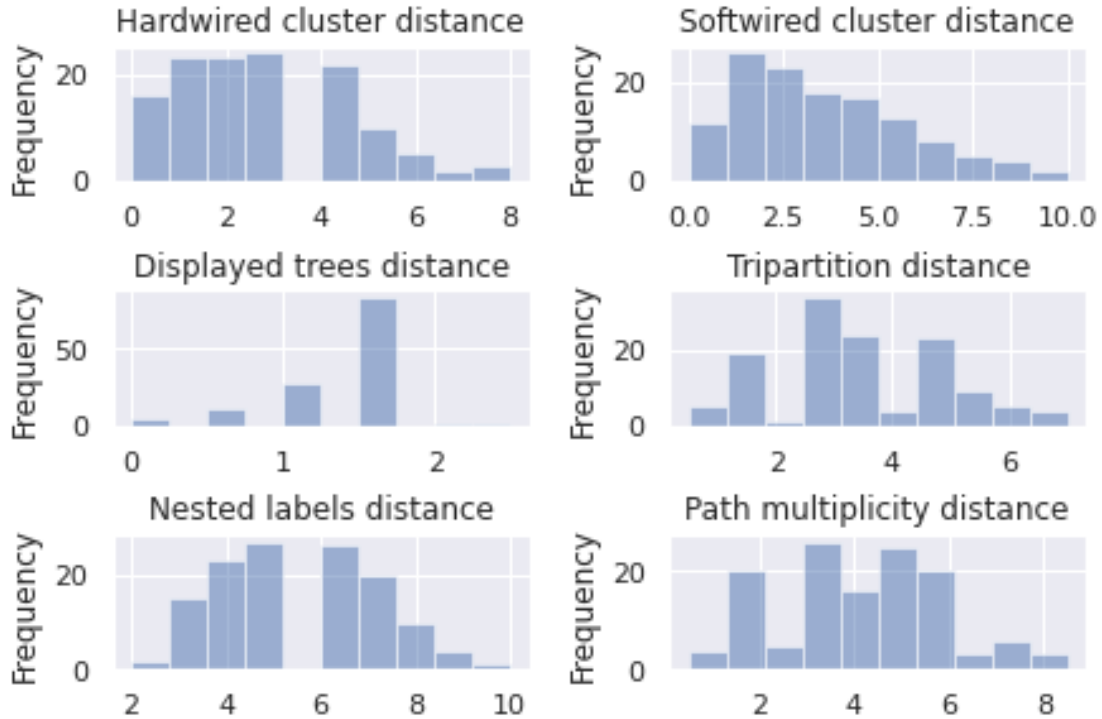
Inferred n\_reticulations less: 100  
Inferred n\_reticulations equal: 28  
Inferred n\_reticulations more: 0



<Figure size 432x288 with 0 Axes>



<Figure size 432x288 with 0 Axes>



### 2.2.2 Plots for LikelihoodType.BEST

```
[20]: df_random_msasize_200_best = df_random_msasize_200.query('likelihood_type == "BEST"')
      build_stats(df_random_msasize_200_best)
```

Inferred BIC better or equal: 124

Inferred BIC worse: 4

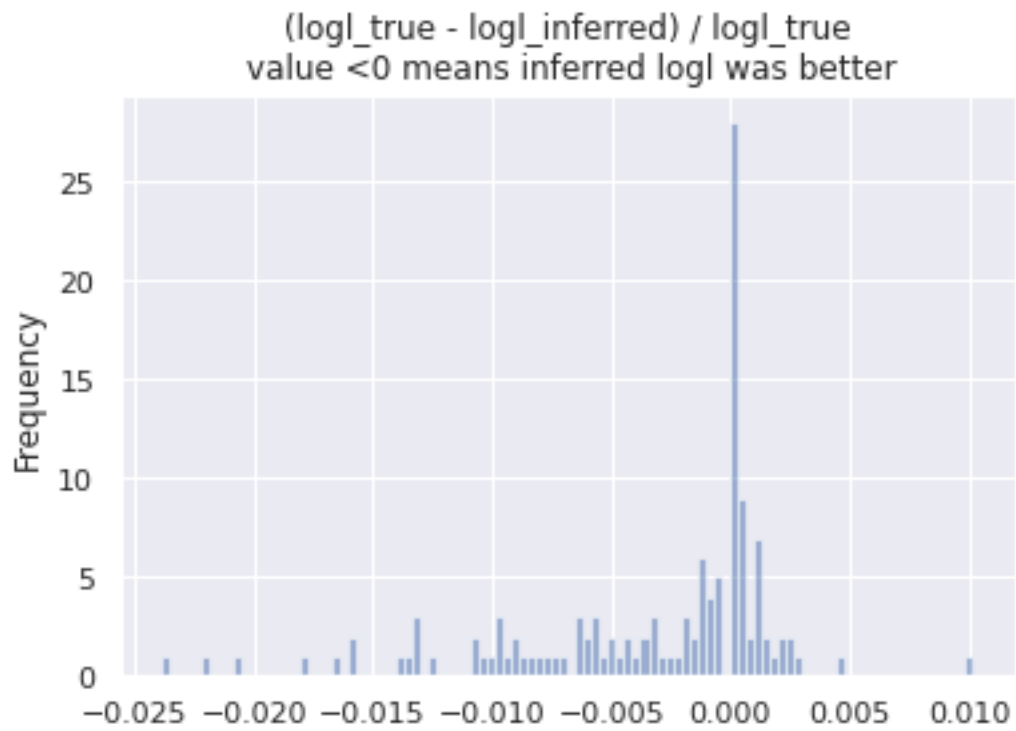
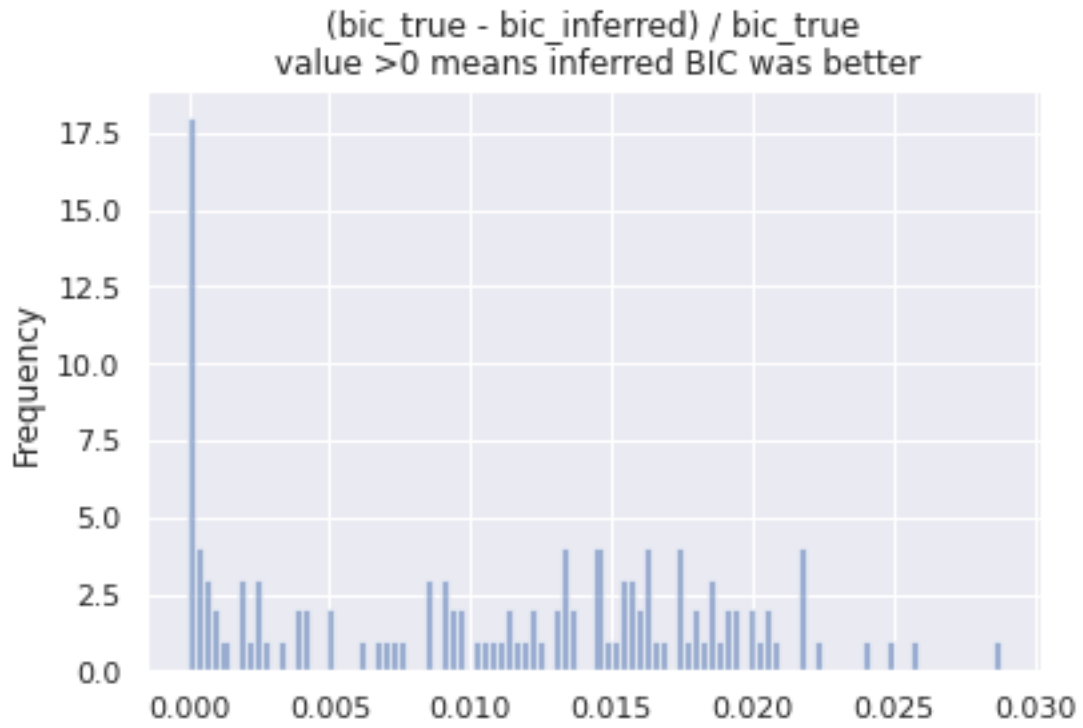
Inferred loglh better or equal: 50

Inferred loglh worse: 78

Inferred n\_reticulations less: 100

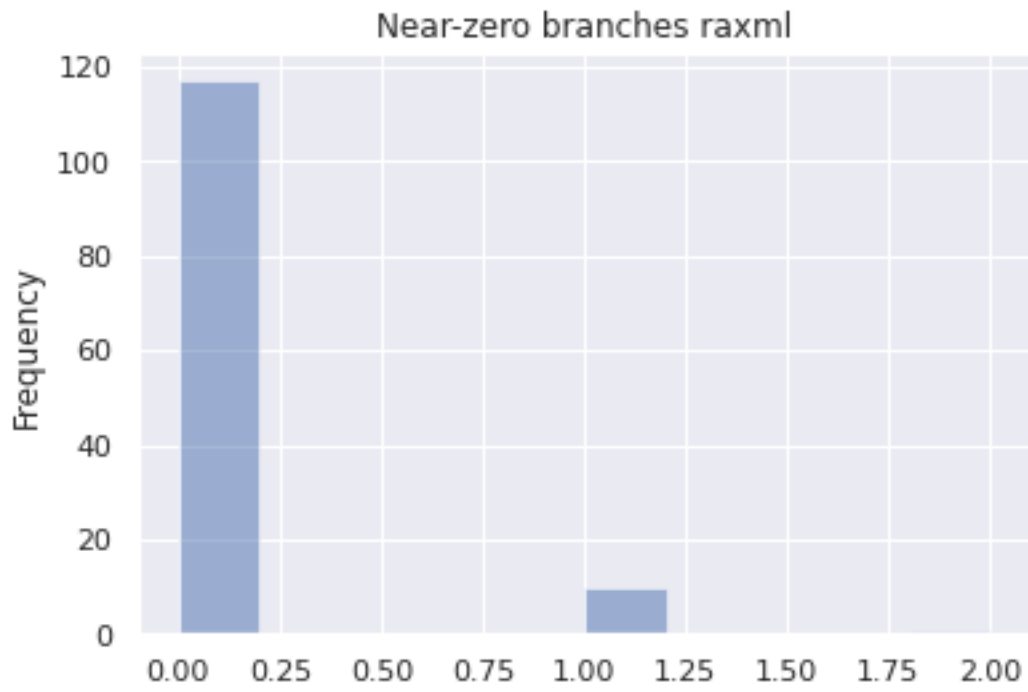
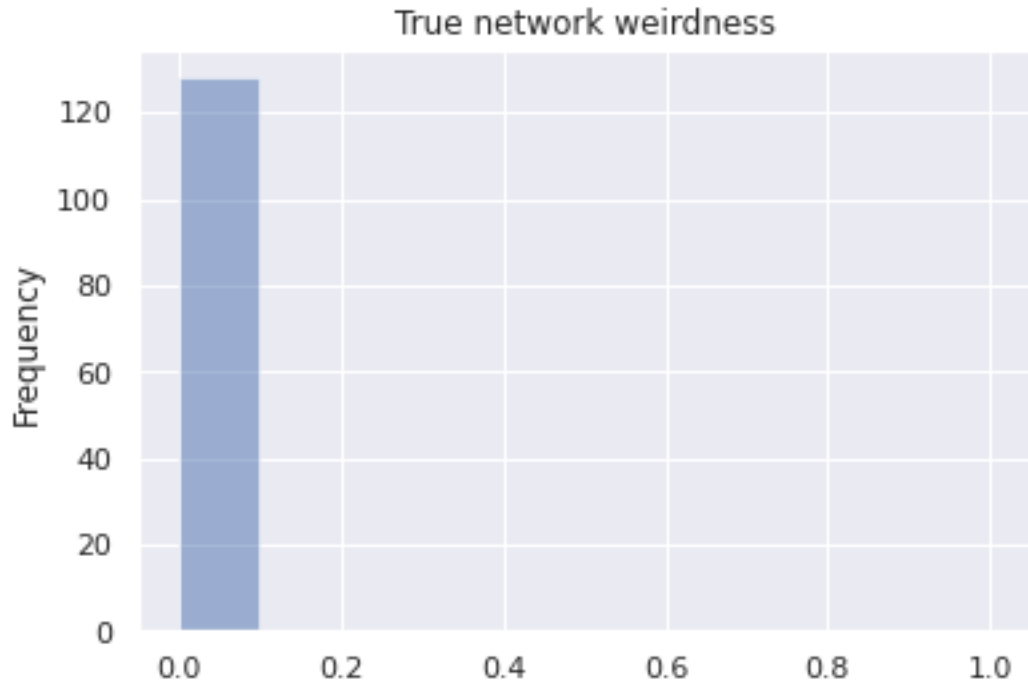
Inferred n\_reticulations equal: 28

Inferred n\_reticulations more: 0





<Figure size 432x288 with 0 Axes>



<Figure size 432x288 with 0 Axes>



[ ]: