

```

In [29]: #data from other script
from pandas import DataFrame
import numpy as np
import matplotlib.pyplot as plt

o1 = {'greedy_cost_u': [14897, 5899640, 14576346980, 32451, 3728, 0, 0, [967399.2680047225, 1037311.5018
'greedy_vote_u': [14897, 5900440, 16390302420, 62481, 3113, 0, 0, [1058380.1023219205, 1196423.355
'mes': [14897, 5694978, 9408283098, 90406, 1873, 0, 0, [567138.0055096418, 719533.7152625152, 6577
'onemin_full': [14897, 5899096, 7270832872, 64837, 937, 0, 0, [512159.9205037387, 482776.698412698
'onemin_agglom': [14897, 5896240, 16388082620, 62326, 3115, 0, 0, [1058339.8819362456, 1196285.186
'bu_1.5_agglom_votes': [14897, 5840573, 5339056883, 18493, 3657, 0, 0, [436066.41873278236, 313384
'bu_1.5_agglom_cost': [14897, 5885186, 6830746980, 34001, 2615, 0, 0, [488041.84179456905, 433100.
'onemin_age': [14897, 5896240, 16388082620, 62326, 3115, 0, 0, [1058339.8819362456, 1196285.186813
'bu_1.5_age_votes': [14897, 5893831, 9068174001, 83476, 2083, 0, 0, [537116.2298307753, 663000.964
'bu_1.5_age_cost': [14897, 5881198, 11699960864, 56936, 2816, 0, 0, [713379.2475403384, 858867.398
'onemin_gend': [14897, 5896240, 16388082620, 62326, 3115, 0, 0, [1058339.8819362456, 1196285.18681
'bu_1.2_gend_votes': [14897, 5886686, 10152434846, 92006, 2059, 0, 0, [613803.5796930343, 773384.1
'bu_1.2_gend_cost': [14897, 5900780, 14058728000, 61857, 3208, 0, 0, [806642.3455332547, 1097395.1

o2 = {'greedy_cost_u': [5045, 2805550, 2876808850, 7249, 1708, 0, 0, [502087.82117163413, 640562.1444201
'greedy_vote_u': [5045, 2806350, 2856904950, 15602, 1192, 0, 0, [502164.08016443986, 653904.048140
'mes': [5045, 2539274, 1218657384, 23488, 721, 0, 0, [217402.55087358685, 275034.6739606127, 24453
'onemin_full': [5045, 2803910, 1330055690, 18256, 338, 0, 0, [268417.38643371017, 237810.733041575
'onemin_agglom': [5045, 2806350, 2856904950, 15602, 1192, 0, 0, [502164.08016443986, 653904.048140
'bu_1.5_agglom_votes': [5045, 2787560, 1331659590, 9216, 826, 0, 0, [279087.838643371, 235672.5492
'bu_1.5_agglom_cost': [5045, 2579935, 1238182800, 7460, 1605, 0, 0, [245722.64542651593, 239297.15
'onemin_age': [5045, 2806350, 2856904950, 15602, 1192, 0, 0, [502164.08016443986, 653904.048140043
'bu_1.5_age_votes': [5045, 2803674, 1429641484, 24460, 844, 0, 0, [251090.3206577595, 320415.41794
'bu_1.5_age_cost': [5045, 2803850, 2630107550, 13640, 1348, 0, 0, [442784.5323741007, 594965.42669
'onemin_gend': [5045, 2806350, 2856904950, 15602, 1192, 0, 0, [502164.08016443986, 653904.04814004
'bu_1.2_gend_votes': [5045, 2675174, 1404043284, 24064, 850, 0, 0, [247221.76978417265, 315354.148
'bu_1.2_gend_cost': [5045, 2805550, 2876808850, 7249, 1708, 0, 0, [502087.82117163413, 640562.1444

o3 = {'greedy_cost_u': [4930, 3084952, 4816903992, 13578, 776, 0, 0, [998254.8811188812, 1050116.8215488
'greedy_vote_u': [4930, 3086092, 5213446532, 28073, 422, 0, 0, [1035654.1818181818, 1090869.144781
'mes': [4930, 3037483, 4223190527, 42064, 132, 0, 0, [832568.3566433566, 886266.0067340067, 881912
'onemin_full': [4930, 3038653, 3471299141, 36566, 83, 0, 0, [713357.9482517482, 726554.2525252525,
'onemin_agglom': [4930, 3083062, 5208449592, 27637, 424, 0, 0, [1034755.2027972027, 1089696.636363
'bu_1.5_agglom_votes': [4930, 610825, 416378175, 3432, 2266, 0, 0, [97161.64335664336, 92049.66329
'bu_1.5_agglom_cost': [4930, 0, 0, 0, 4930, 7, 2, [0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [0.0, 0.0,
'onemin_age': [4930, 3083062, 5208449592, 27637, 424, 0, 0, [1034755.2027972027, 1089696.636363636
'bu_1.5_age_votes': [4930, 3086388, 4297519412, 42331, 315, 0, 0, [838898.6293706293, 907907.11784
'bu_1.5_age_cost': [4930, 3086603, 4942671188, 28826, 421, 0, 0, [1011605.2265734266, 1022866.7946
'onemin_gend': [4930, 3083062, 5208449592, 27637, 424, 0, 0, [1034755.2027972027, 1089696.63636363
'bu_1.2_gend_votes': [4930, 3086388, 4297519412, 42331, 315, 0, 0, [838898.6293706293, 907907.1178
'bu_1.2_gend_cost': [4930, 3087462, 5224268592, 26558, 438, 0, 0, [1033136.4615384615, 1092480.643

## we've replaced MES in the code above

# forming result_list

result_list = [o1, o2, o3]

```

In [32]: *##running the sorts on the result List*

```
print("_____")
print("Results by Algorithm")

algwise_utilities = {}
algwise_worst_group_ratio = {}
algwise_uncovered_people = {}

for alg in result_list[0].keys():
    utilities = []
    uncovered_people = []
    worst_group_ratio = []

    for res in result_list:
        try:
            utilities.append(res[alg][2:4])
            uncovered_people.append([res[alg][4] / res[alg][0]])
            worst_group_ratio.append([min(res[alg][7]) / (res[alg][2] / res[alg][0]),
                                      min(res[alg][8]) / (res[alg][3] / res[alg][0]),
                                      min(res[alg][9]) / (res[alg][2] / res[alg][0]),
                                      min(res[alg][10]) / (res[alg][3] / res[alg][0])])

        except:
            print(f"alg {alg} had failed on result index: {result_list.index(res)}")
    algwise_utilities[alg] = utilities
    algwise_worst_group_ratio[alg] = worst_group_ratio
    algwise_uncovered_people[alg] = uncovered_people

print("Algorithm Utilities")
print(algwise_utilities)
print("_____")
print("Algorithm Worst Group Ratio of Utilities")
print(algwise_worst_group_ratio)
print("_____")
print("Algorithm Percent of People Uncovered")
print(algwise_uncovered_people)
print("_____")
print("DONE")
```

Results by Algorithm

alg bu_1.5_agglom_cost had failed on result index: 2

Algorithm Utilities

```
{'greedy_cost_u': [[14576346980, 32451], [2876808850, 7249], [4816903992, 13578]], 'greedy_vote_u': [[16390302420, 62481], [2856904950, 15602], [5213446532, 28073]], 'mes': [[9408283098, 90406], [1218657384, 23488], [4223190527, 42064]], 'onemin_full': [[7270832872, 64837], [1330055690, 18256], [3471299141, 36566]], 'onemin_agglom': [[16388082620, 62326], [2856904950, 15602], [5208449592, 27637]], 'bu_1.5_agglom_votes': [[5339056883, 18493], [1331659590, 9216], [416378175, 3432]], 'bu_1.5_agglom_cost': [[6830746980, 34001], [1238182800, 7460], [0, 0]], 'onemin_age': [[16388082620, 62326], [2856904950, 15602], [5208449592, 27637]], 'bu_1.5_age_votes': [[9068174001, 83476], [1429641484, 24460], [4297519412, 42331]], 'bu_1.5_age_cost': [[11699960864, 56936], [2630107550, 13640], [4942671188, 28826]], 'onemin_gen_d': [[16388082620, 62326], [2856904950, 15602], [5208449592, 27637]], 'bu_1.2_gend_votes': [[10152434846, 92006], [1404043284, 24064], [4297519412, 42331]], 'bu_1.2_gend_cost': [[14058728000, 61857], [2876808850, 7249], [5224268592, 26558]]}
```

Algorithm Worst Group Ratio of Utilities

```
{'greedy_cost_u': [[0.4840021426006414, 0.5544507263098369, 0.9969028779587851, 0.9382860064499075], [0.4766947962186081, 0.592304734622237, 0.9245061229180523, 0.922283070779374], [0.506359982430701, 0.5285448724761388, 0.9815587375531267, 0.9730962524351597]], 'greedy_vote_u': [[0.47894610651473307, 0.5903844292474436, 0.9989658011891386, 0.911648879063091], [0.5431861728397535, 0.6467119600051275, 0.9157040202657326, 0.8808551845687369], [0.5894363574184099, 0.6568837032925461, 0.9547144243854433, 0.9415781968526397]], 'mes': [[0.5691436804772864, 0.6491289462552432, 0.945750190935685, 0.8985905103621311], [0.8302998712577885, 0.7723319105455388, 0.9313389197735706, 0.8945683334441525], [0.734173381577384, 0.8159657857105026, 0.9436621162330985, 0.9258151701790847]], 'onemin_full': [[0.827739685695866, 0.7320516777771123, 0.9989590060536243, 0.9131883796391288], [0.7928562751788583, 0.7702451656814664, 0.9530663615195342, 0.9073501404111861], [0.613770084096651, 0.7782286512268088, 0.9740092235961293, 0.933074707816616]], 'onemin_agglom': [[0.4787355219126767, 0.5773667656990829, 0.9990157130976219, 0.9131948205497316], [0.5431861728397535, 0.6467119600051275, 0.9157040202657326, 0.8808551845687369], [0.5894819799038421, 0.6661176564347099, 0.9544802085955439, 0.933839596327494]], 'bu_1.5_agglom_votes': [[0.6763735823530583, 0.767188614482185, 0.9518412365501081, 0.9824042606842994], [0.5438366537594944, 0.564888307106974, 0.9370088529982051, 0.9718842187672919], [0.8172584204737753, 0.8876161152280555, 0.9888306935591692, 0.9564356887602056]], 'bu_1.5_agglom_cost': [[0.7302443405013792, 0.8231611726214826, 0.9913831165731574, 0.8993319041723368], [0.5444205680274935, 0.5755518795276938, 0.9434191685321688, 0.9403164820610321]], 'onemin_age': [[0.4787355219126767, 0.5773667656990829, 0.9990157130976219, 0.9131948205497316], [0.5431861728397535, 0.6467119600051275, 0.9157040202657326, 0.8808551845687369], [0.5894819799038421, 0.6661176564347099, 0.9544802085955439, 0.933839596327494]], 'bu_1.5_age_votes': [[0.7996112434682877, 0.7485985551177942, 0.9464494608723935, 0.9037209183042085], [0.843641605693403, 0.7767479688940694, 0.8940782142795607, 0.8801793032286227], [0.6929332539630615, 0.7857574527929228, 0.9452373140255442, 0.9281505803713805]], 'bu_1.5_age_cost': [[0.7316405770737334, 0.7373622057301978, 0.9447831729599494, 0.9005137869834821], [0.7611961513440205, 0.8656485929993137, 0.9006945713136241, 0.8742000473180591], [0.7058878958697622, 0.8605430048646309, 0.9428687387919725, 0.9021934250467198]], 'onemin_gend': [[0.4787355219126767, 0.5773667656990829, 0.9990157130976219, 0.9131948205497316], [0.5431861728397535, 0.6467119600051275, 0.9157040202657326, 0.8808551845687369], [0.5894819799038421, 0.6661176564347099, 0.9544802085955439, 0.933839596327494]], 'bu_1.2_gend_votes': [[0.569755138058831, 0.6518589555565405, 0.9499054728702317, 0.9004173950607361], [0.8427577519011478, 0.7739180447883657, 0.8974043378995705, 0.8832832731573705], [0.6929332539630615, 0.7857574527929228, 0.9452373140255442, 0.9281505803713805]], 'bu_1.2_gend_cost': [[0.5036012761911491, 0.5994677571863662, 0.9709078309118135, 0.9129321324509098], [0.4766947962186081, 0.592304734622237, 0.9245061229180523, 0.922283070779374], [0.584077634947201, 0.6532347162789633, 0.9578133064109895, 0.9432645813991097]]}
```

Algorithm Percent of People Uncovered

```
{'greedy_cost_u': [[0.25025172853594685], [0.3385530227948464], [0.15740365111561866]], 'greedy_vote_u': [[0.2089682486406659], [0.2362735381565907], [0.08559837728194726]], 'mes': [[0.12573001275424583], [0.14291377601585728], [0.026774847870182555]], 'onemin_full': [[0.06289857018191582], [0.0669970267591675], [0.016835699797160243]], 'onemin_agglom': [[0.20910250385983756], [0.2362735381565907], [0.08600405679513184]], 'bu_1.5_agglom_votes': [[0.24548566825535342], [0.1637264618434093], [0.4596348884381339]], 'bu_1.5_agglom_cost': [[0.17553869906692623], [0.31813676907829536], [1.0]], 'onemin_age': [[0.20910250385983756], [0.2362735381565907], [0.08600405679513184]], 'bu_1.5_age_votes': [[0.13982681076726858], [0.16729435084241823], [0.06389452332657201]], 'bu_1.5_age_cost': [[0.18903134859367657], [0.267195242814668], [0.08539553752535498]], 'onemin_gend': [[0.20910250385983756], [0.2362735381565907], [0.08600405679513184]], 'bu_1.2_gend_votes': [[0.13821574813720883], [0.16848364717542122], [0.06389452332657201]], 'bu_1.2_gend_cost': [[0.21534537155131905], [0.3385530227948464], [0.08884381338742393]]}
```

DONE

```

In [40]: ##plot utilities

keys = algwise_utilities.keys()

print("cost utilities by algorithm")
for key in keys:
    print(f"{key}: {[x[0] for x in algwise_utilities[key]]}")

print("vote utilities by algorithm")
for key in keys:
    print(f"{key}: {[x[1] for x in algwise_utilities[key]]}")

cu_table = [[],[],[]]
vu_table = [[],[],[]]

for key in keys:
    for i in range(3):
        cu_table[i].append(algwise_utilities[key][i][0])
        vu_table[i].append(algwise_utilities[key][i][1])

cu = np.array(cu_table)
vu = np.array(vu_table)
print(cu)
print(vu)

dfcu = DataFrame(cu, columns=keys, index=[1,2,3])
dfvu = DataFrame(vu, columns=keys, index=[1,2,3])

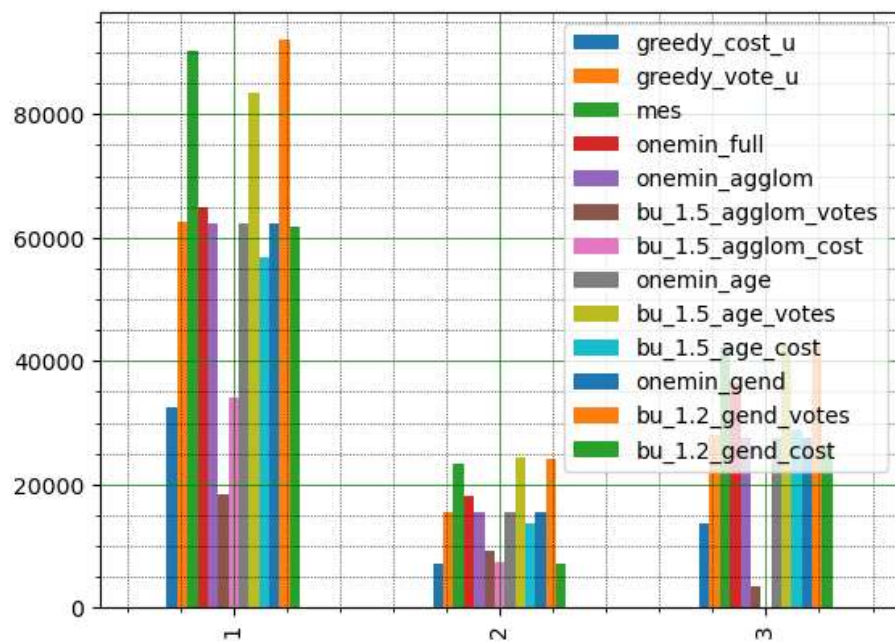
dfcu.plot(kind='bar')
# Turn on the grid
plt.minorticks_on()
plt.grid(which='major', linestyle='-', linewidth='0.5', color='green')
plt.grid(which='minor', linestyle=':', linewidth='0.5', color='black')
plt.legend(loc = 1)

plt.show()

dfvu.plot(kind='bar')
# Turn on the grid
plt.minorticks_on()
plt.grid(which='major', linestyle='-', linewidth='0.5', color='green')
plt.grid(which='minor', linestyle=':', linewidth='0.5', color='black')
plt.legend(loc = 1)

plt.show()

```


```
worst_bounds = []

for key in keys:
    t = algwise_worst_group_ratio[key]
    lowest = t[0]
    for x in t[1:3]:
        for i in range(len(t)):
            if x[i] < lowest[i]:
                lowest[i] = x[i]
    worst_bounds.append(lowest)

print(worst_bounds)

wb = np.transpose(np.asarray(worst_bounds))

dfwb = DataFrame(wb, columns = keys, index = [1,2,3,4])

dfwb.plot(kind='bar')
# Turn on the grid
plt.minorticks_on()
plt.grid(which='major', linestyle='-', linewidth='0.5', color='green')
plt.grid(which='minor', linestyle=':', linewidth='0.5', color='black')
plt.legend(loc = 1)

plt.show()
```

Bar chart showing the performance of various methods across four categories (1, 2, 3, 4). The y-axis represents a performance metric from 0.0 to 1.0. The legend lists 14 methods: greedy_cost_u, greedy_vote_u, mes, onemin_full, onemin_agglom, bu_1.5_agglom_votes, bu_1.5_agglom_cost, onemin_age, bu_1.5_age_votes, bu_1.5_age_cost, onemin_gend, bu_1.2_gend_votes, and bu_1.2_gend_cost. Methods 1-12 are grouped by color, while methods 13 and 14 are shown in grey.

```

In [46]: uc = []

for key in keys:
    uc.append([val[0] for val in pairwise_uncovered_people[key]])

print(uc)

uncov = np.transpose(np.asarray(uc))

dfuc = DataFrame(uncov, columns = keys, index = [1,2,3])

dfuc.plot(kind='bar')
# Turn on the grid
plt.minorticks_on()
plt.grid(which='major', linestyle='-', linewidth='0.5', color='green')
plt.grid(which='minor', linestyle=':', linewidth='0.5', color='black')
plt.legend(loc = 1)

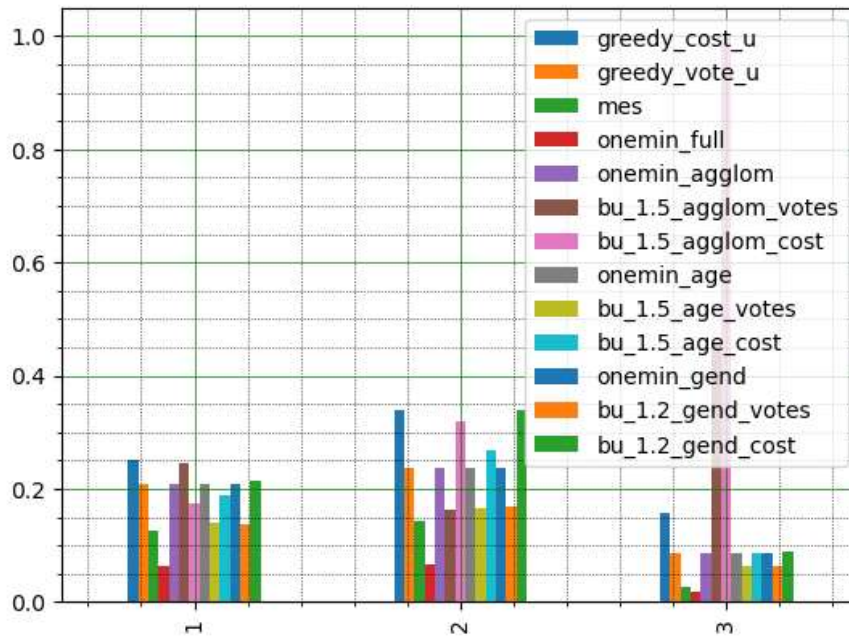
plt.show()

```

```

[[0.25025172853594685, 0.3385530227948464, 0.15740365111561866], [0.2089682486406659, 0.236273538156590
7, 0.08559837728194726], [0.12573001275424583, 0.14291377601585728, 0.026774847870182555], [0.062898570
18191582, 0.0669970267591675, 0.016835699797160243], [0.20910250385983756, 0.2362735381565907, 0.086004
05679513184], [0.24548566825535342, 0.1637264618434093, 0.4596348884381339], [0.17553869906692623, 0.31
813676907829536, 1.0], [0.20910250385983756, 0.2362735381565907, 0.08600405679513184], [0.1398268107672
6858, 0.16729435084241823, 0.06389452332657201], [0.18903134859367657, 0.267195242814668, 0.08539553752
535498], [0.20910250385983756, 0.2362735381565907, 0.08600405679513184], [0.13821574813720883, 0.168483
64717542122, 0.06389452332657201], [0.21534537155131905, 0.3385530227948464, 0.08884381338742393]]

```



In []: