Visualization

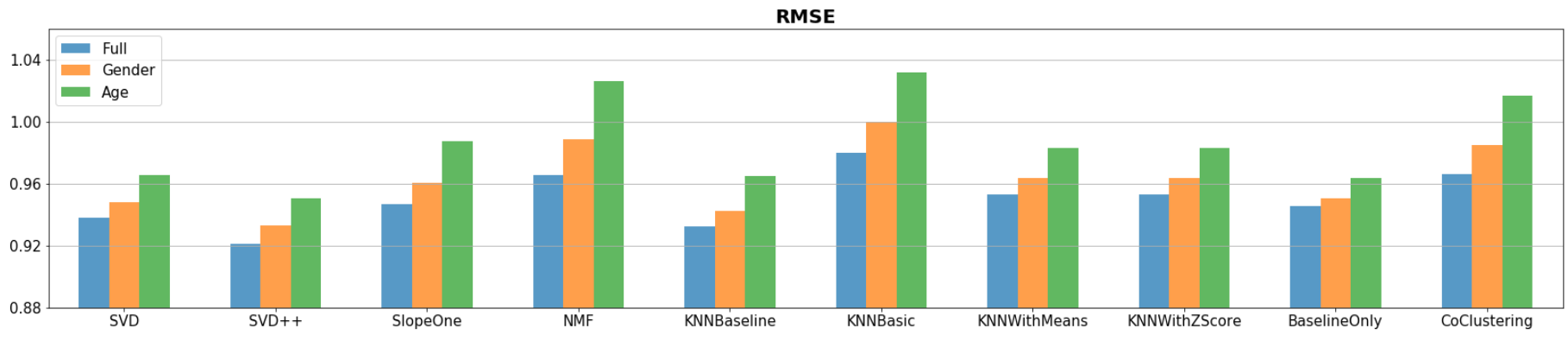
In [1]:
```
import numpy as np
import matplotlib.pyplot as plt

%matplotlib inline
```

traditional algos ml-100k

In [237]:
```
data = {
    'SVD':{
        'RMSE':   {'Full':0.9382, 'Gender':0.9481, 'Age':0.9657}
    },
    'SVD++':{
        'RMSE':   {'Full':0.9212, 'Gender':0.9333, 'Age':0.9504}
    },
    'SlopeOne':{
        'RMSE':   {'Full':0.9470, 'Gender':0.9603, 'Age':0.9871}
    },
    'NMF':{
        'RMSE':   {'Full':0.9655, 'Gender':0.9888, 'Age':1.0259}
    },
    'KNNBaseline':{
        'RMSE':   {'Full':0.9323, 'Gender':0.9426, 'Age':0.9651}
    },
    'KNNBasic':{
        'RMSE':   {'Full':0.9801, 'Gender':1.0001, 'Age':1.0320}
    },
    'KNNWithMeans':{
        'RMSE':   {'Full':0.9527, 'Gender':0.9638, 'Age':0.9832}
    },
    'KNNWithZScore':{
        'RMSE':   {'Full':0.9527, 'Gender':0.9637, 'Age':0.9832}
    },
    'BaselineOnly':{
        'RMSE':   {'Full':0.9457, 'Gender':0.9505, 'Age':0.9639}
    },
    'CoClustering':{
        'RMSE':   {'Full':0.9660, 'Gender':0.9849, 'Age':1.0165}
    },
}
```

In [238]:
```
def get_x_data(size=10, total_width=1.2):
    """
    return x_full, x_gender, x_age
    """
    x = np.arange(1,21,2)
    n=3
    width = total_width / n
    x = x - (total_width - width) / 2
    return x, x+width, x+2*width, width
```

In [242]:
```
# Vizualization
plt.figure(figsize=(25,5))
plt.subplots_adjust(left=0.075, right=0.95, bottom=0.1, top=0.9, wspace=0.2, hspace=0.2)
fontsize = 20
fontsize_axis = 15
size = 10 # number of algos
total_width=1.2
x_full, x_gender, x_age, width = get_x_data(size)
xticks = list(data.keys())

plt.title("RMSE", fontsize=fontsize, weight='bold')
plt.xlim(0,20)
plt.xticks(list(range(1,21,2)), xticks, fontsize=fontsize_axis)
plt.ylim(0.88,1.06)
plt.yticks([0.88, 0.92, 0.96, 1.00, 1.04], fontsize=fontsize_axis)
plt.grid(axis='y')

# build y
y_full = []
y_gender = []
y_age = []
for key in data.keys():
    y_full.append(data[key]['RMSE']['Full'])
    y_gender.append(data[key]['RMSE']['Gender'])
    y_age.append(data[key]['RMSE']['Age'])

plt.bar(x_full, y_full,  width=width, label='Full', color="tab:blue", alpha=0.75)
plt.bar(x_gender, y_gender, width=width, label='Gender', color="tab:orange",alpha=0.75)
plt.bar(x_age, y_age, width=width, label='Age', color="tab:green",alpha=0.75)
plt.legend(fontsize=fontsize_axis, loc='upper left')


# save
# plt.savefig("figs/ml-1m_noDL.png", dpi=600)
plt.savefig("figs/ml-100k_noDL.pdf")
plt.show()
```



In [ ]:
```
```

traditional algos

- 2个大柱状图：rmse / recall
- 每个坐标一种传统算法，每个传统算法对比全数据，分开考虑性别，以及分开考虑年龄

In [215]:
```
data = {
    'SVD':{
        'RMSE':   {'Full':0.8735, 'Gender':0.8829, 'Age':0.9031},
        'Recall': {'Full':0.5534, 'Gender':0.5503, 'Age':0.5420}
    },
    'SlopeOne':{
        'RMSE':   {'Full':0.9055, 'Gender':0.9057, 'Age':0.9176},
        'Recall': {'Full':0.5399, 'Gender':0.5385, 'Age':0.5346}
    },
    'NMF':{
        'RMSE':   {'Full':0.9151, 'Gender':0.9169, 'Age':0.9336},
        'Recall': {'Full':0.5281, 'Gender':0.5262, 'Age':0.5224}
    },
    # 'NormalPredictor':{
    #     'RMSE':   {'Full':1.5034, 'Gender':, 'Age':},
    #     'Recall': {'Full':0.3963, 'Gender':, 'Age':}
    # },
    'KNNBaseline':{
        'RMSE':   {'Full':0.8936, 'Gender':0.8939, 'Age':0.9029},
        'Recall': {'Full':0.5537, 'Gender':0.5544, 'Age':0.5537}
    },
    'KNNBasic':{
        'RMSE':   {'Full':0.9219, 'Gender':0.9250, 'Age':0.9444},
        'Recall': {'Full':0.5774, 'Gender':0.5742, 'Age':0.5675}
    },
    'KNNWithMeans':{
        'RMSE':   {'Full':0.9277, 'Gender':0.9257, 'Age':0.9283},
        'Recall': {'Full':0.5171, 'Gender':0.5221, 'Age':0.5285}
    },
    'KNNWithZScore':{
        'RMSE':   {'Full':0.9292, 'Gender':0.9259, 'Age':0.9273},
        'Recall': {'Full':0.5215, 'Gender':0.5256, 'Age':0.5326}
    },
    'BaselineOnly':{
        'RMSE':   {'Full':0.9075, 'Gender':0.9075, 'Age':0.9141},
        'Recall': {'Full':0.5525, 'Gender':0.5523, 'Age':0.5490}
    },
    'CoClustering':{
        'RMSE':   {'Full':0.9135, 'Gender':0.9190, 'Age':0.9367},
        'Recall': {'Full':0.5470, 'Gender':0.5448, 'Age':0.5407}
    },
}
```

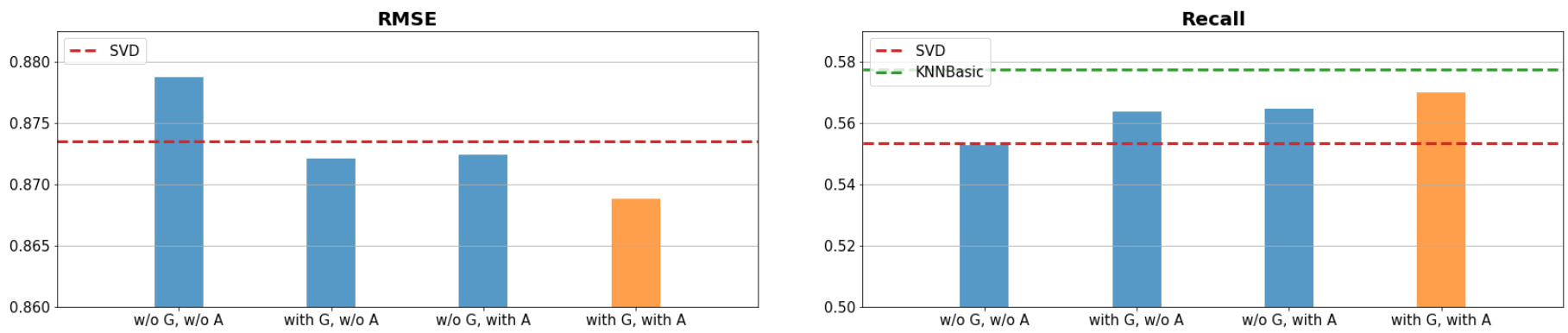
In [216]:
```
def get_x_data(size=9, total_width=1.2):
    """
    return x_full, x_gender, x_age
    """
    x = np.arange(1,19,2)
    n=3
    width = total_width / n
    x = x - (total_width - width) / 2
    return x, x+width, x+2*width, width
```

In [218]:
```
# Vizualization
plt.figure(figsize=(25,10))
plt.subplots_adjust(left=0.075, right=0.95, bottom=0.1, top=0.9, wspace=0.2, hspace=0.2)
fontsize = 20
fontsize_axis = 15
size = 9 # number of algos
total_width=1.2
x_full, x_gender, x_age, width = get_x_data(size)
xticks = list(data.keys())

plt.subplot(211)
plt.title("RMSE", fontsize=fontsize, weight='bold')
plt.xlim(0,18)
plt.xticks(list(range(1,19,2)), xticks, fontsize=fontsize_axis)
plt.ylim(0.86,0.95)
plt.yticks([0.86, 0.88, 0.90, 0.92, 0.94], fontsize=fontsize_axis)
plt.grid(axis='y')

# build y
y_full = []
y_gender = []
y_age = []
for key in data.keys():
    y_full.append(data[key]['RMSE']['Full'])
    y_gender.append(data[key]['RMSE']['Gender'])
    y_age.append(data[key]['RMSE']['Age'])

plt.bar(x_full, y_full,  width=width, label='Full', color="tab:blue", alpha=0.75)
plt.bar(x_gender, y_gender, width=width, label='Gender', color="tab:orange",alpha=0.75)
plt.bar(x_age, y_age, width=width, label='Age', color="tab:green",alpha=0.75)
plt.legend(fontsize=fontsize_axis, loc='upper left')


plt.subplot(212)
plt.title("Recall", fontsize=fontsize, weight='bold')
plt.xlim(0,18)
plt.xticks(list(range(1,19,2)), xticks, fontsize=fontsize_axis)
plt.ylim(0.5,0.59)
plt.yticks([0.50, 0.52, 0.54, 0.56, 0.58], fontsize=fontsize_axis)
plt.grid(axis='y')

# build y
y_full = []
y_gender = []
y_age = []
for key in data.keys():
    y_full.append(data[key]['Recall']['Full'])
    y_gender.append(data[key]['Recall']['Gender'])
    y_age.append(data[key]['Recall']['Age'])

plt.bar(x_full, y_full,  width=width, label='Full', color="tab:blue",alpha=0.75)
plt.bar(x_gender, y_gender, width=width, label='Gender', color="tab:orange",alpha=0.75)
plt.bar(x_age, y_age, width=width, label='Age', color="tab:green",alpha=0.75)
plt.legend(fontsize=fontsize_axis, loc='upper left')


# save
# plt.savefig("figs/ml-1m_noDL.png", dpi=600)
# plt.savefig("figs/ml-1m_noDL.pdf")
plt.show()
```

dl algos -1

- dl主算法
- 两个柱状图(左右): precision / recall
- 对比
  - w/o G, w/o A: tab:blue
  - with G, w/o A: tab:orange
  - w/o G, with A: tab:green
  - with G, with A: tab:red
  - 两条虚线: non-DL method的最佳水平: tab:brown

In [221]:
```
plt.figure(figsize=(25,5))
fontsize = 20
fontsize_axis = 15
plt.subplots_adjust(left=0.075, right=0.95, bottom=0.1, top=0.9, wspace=0.15, hspace=0.2)
xticks = ['w/o G, w/o A', 'with G, w/o A', 'w/o G, with A', 'with G, with A']
x = [1.25, 2.5, 3.75, 5]

plt.subplot(121)
plt.title("RMSE", fontsize=fontsize, weight='bold')
plt.xlim(0.25,6)
plt.xticks(x, xticks, fontsize=fontsize_axis)
plt.ylim(0.86,0.8825)
plt.yticks([0.86, 0.865, 0.87, 0.875, 0.88], fontsize=fontsize_axis)
plt.grid(axis='y')
y = [0.8787, 0.8721, 0.8724, 0.8688]
plt.bar(x[:-1],y[:-1], color = "tab:blue", alpha=0.75, width=0.4)
plt.bar(x[-1:],y[-1:], color = "tab:orange", alpha=0.75, width=0.4)
# SVD 0.8735
plt.hlines(0.8735, 0, 6, colors = "tab:red", linestyles = "--", label="SVD",linewidth=3)
plt.legend(fontsize=fontsize_axis, loc='upper left')


plt.subplot(122)
plt.title("Recall", fontsize=fontsize, weight='bold')
plt.xlim(0.25,6)
plt.xticks(x, xticks, fontsize=fontsize_axis)
plt.ylim(0.5,0.59)
plt.yticks([0.50, 0.52, 0.54, 0.56, 0.58], fontsize=fontsize_axis)
plt.grid(axis='y')
y = [0.5528, 0.5637, 0.5647, 0.5698]
plt.bar(x[:-1],y[:-1], color = "tab:blue", alpha=0.75, width=0.4)
plt.bar(x[-1:],y[-1:], color = "tab:orange", alpha=0.75, width=0.4)
plt.hlines(0.5534, 0, 6, colors = "tab:red", linestyles = "--", label="SVD",linewidth=3)
plt.hlines(0.5774, 0, 6, colors = "tab:green", linestyles = "--", label="KNNBasic",linewidth=3)
plt.legend(fontsize=fontsize_axis, loc='upper left')

# plt.savefig("figs/ml-1m_DL-1.png", dpi=600)
# plt.savefig("figs/ml-1m_DL-1.pdf")

plt.show()
```



DL algos 2: 各种调整参数细节

- 均为上rmse, 下recall
  - n_um: user / movie的embed dim
  - n_gender: 性别的embed dim
  - n_age: 年龄的embed dim
  - hidden: hidden layer dim
- w/o G, w/o A
  - 保持n_um, 变换hidden
  - 保持hidden dim变换n_factor
- with G, w/o A
  - 保持user/movie dim变换hidden
  - 保持hidden变换n_factor
  - 变换g_factor
- w/o G, with A
  - 保持n_factor变换hidden
  - 保持hidden变换n_factor
  - 变换a_factor
- with G, with A: 这里n/h不变 --> 8行3列
  - 变换g_factor
  - 变换a_factor

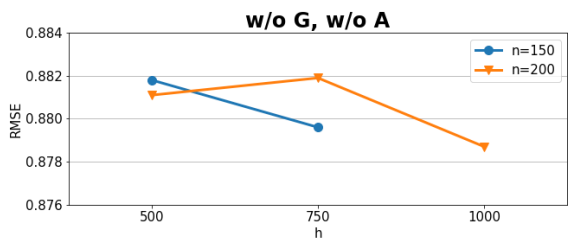
In [235]:
```
# vis --> 8
plt.figure(figsize=(32,32))
fontsize = 24
fontsize_axis = 15
ms = 10
linewidth = 3
plt.subplots_adjust(left=0.075, right=0.95, bottom=0.05, top=0.95, wspace=0.25, hspace=0.35)

# ---------------------------------------------------  w/o G, w/o A
plt.subplot(831) # rmse
plt.title("w/o G, w/o A", fontsize=fontsize, weight='bold')
plt.xlabel("h", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[500, 750, 1000], fontsize=fontsize_axis)
plt.ylabel("RMSE", fontsize=fontsize_axis)
plt.ylim(0.876, 0.884)
plt.yticks([0.876, 0.878, 0.880, 0.882, 0.884], fontsize=fontsize_axis)
plt.grid(axis='y')
# n_um = 150, hidden = [500, 1000]
plt.plot([1,2], [0.8818, 0.8796] , label="n=150", marker="o", linewidth=linewidth,ms=ms)
# n_um = 200, hidden = [500, 750, 1000]
plt.plot([1,2,3], [0.8811, 0.8819, 0.8787], label="n=200", marker="v", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)

plt.subplot(832) # rmse
plt.title("w/o G, w/o A", fontsize=fontsize, weight='bold')
plt.xlabel("n", fontsize=fontsize_axis)
plt.xlim(0.5,4.5)
plt.xticks([1,2,3,4],[100,150,200,250], fontsize=fontsize_axis)
plt.ylabel("RMSE", fontsize=fontsize_axis)
plt.ylim(0.876, 0.884)
plt.yticks([0.876, 0.878, 0.880, 0.882, 0.884], fontsize=fontsize_axis)
plt.grid(axis='y')
# hidden = 500, n_um = [150, 200]
plt.plot([2,3], [0.8818, 0.8819] , label="h=500", marker="o", linewidth=linewidth,ms=ms)
# hidden = 1000, n_um = [100, 150, 200, 250]
plt.plot([1,2,3,4], [0.8795, 0.8796, 0.8787, 0.8800], label="h=1000", marker="v", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)

plt.subplot(834) # recall
plt.xlabel("h", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[500, 750, 1000], fontsize=fontsize_axis)
plt.ylabel("Recall", fontsize=fontsize_axis)
plt.ylim(0.55, 0.566)
plt.yticks([0.55, 0.554, 0.558, 0.562, 0.566], fontsize=fontsize_axis)
plt.grid(axis='y')
# n_um = 150, hidden = [500, 1000]
plt.plot([1,2], [0.5582, 0.5613] , label="n=150", marker="o", linewidth=linewidth,ms=ms)
# n_um = 200, hidden = [500, 750, 1000]
plt.plot([1,2,3],[0.5599, 0.5597, 0.5528] , label="n=200", marker="v", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)

plt.subplot(835) # recall
plt.xlabel("n", fontsize=fontsize_axis)
plt.xlim(0.5,4.5)
plt.xticks([1,2,3,4],[100,150,200,250], fontsize=fontsize_axis)
plt.ylabel("Recall", fontsize=fontsize_axis)
plt.ylim(0.55, 0.566)
plt.yticks([0.55, 0.554, 0.558, 0.562, 0.566], fontsize=fontsize_axis)
plt.grid(axis='y')
# hidden = 500, n_um = [150, 200]
plt.plot([2,3], [0.5582, 0.5597] , label="h=500", marker="o", linewidth=linewidth,ms=ms)
# hidden = 1000, n_um = [100, 150, 200, 250]
plt.plot([1,2,3,4], [0.5630, 0.5613, 0.5528, 0.5558], label="h=1000", marker="v", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)


# ------------------------------------------------------------------ with G, w/o A
plt.subplot(837) # rmse
plt.title("with G, w/o A", fontsize=fontsize, weight='bold')
plt.xlabel("h", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[500, 750, 1000], fontsize=fontsize_axis)
plt.ylabel("RMSE", fontsize=fontsize_axis)
plt.ylim(0.871, 0.879)
plt.yticks([0.871, 0.873, 0.875, 0.877, 0.879], fontsize=fontsize_axis)
plt.grid(axis='y')
# n100,g25,h=[500, 750, 1000]:[0.8743,0.5572],[0.8728,0.5610],[0.8725,0.5622]
# n200,g25,h=[500, 750, 1000]:[0.8739,0.5617],[0.8722,0.5599],[0.8721,0.5637]
# n200,g50,h=[500, 750, 1000]:[0.8750,0.5564],[0.8738,0.5639],[0.8735,0.5608]
plt.plot([1,2,3], [0.8743,0.8728,0.8725] , label="n=150,g=25", marker="o", linewidth=linewidth,ms=ms)
plt.plot([1,2,3], [0.8739,0.8722,0.8721] , label="n=200,g=25", marker="v", linewidth=linewidth,ms=ms)
plt.plot([1,2,3], [0.8750,0.8738,0.8735] , label="n=200,g=50", marker="s", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)


plt.subplot(8,3,8) # rmse
```

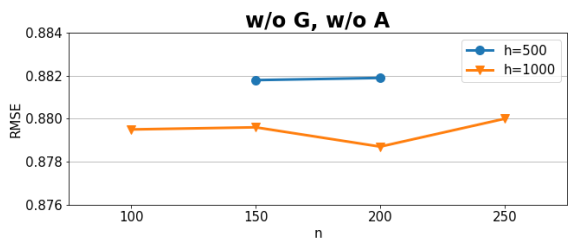
```
plt.title("with G, w/o A", fontsize=fontsize, weight='bold')
plt.xlabel("n", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[100, 150, 200], fontsize=fontsize_axis)
plt.ylabel("RMSE", fontsize=fontsize_axis)
plt.ylim(0.871, 0.879)
plt.yticks([0.871, 0.873, 0.875, 0.877, 0.879], fontsize=fontsize_axis)
plt.grid(axis='y')
# h500,g25,n=[100,150,200]:[0.8743,0.5572],[0.8747,0.5624],[0.8739,0.5617]
# h500,g50,n=[150,200]:[0.8744,0.5600],[0.8750,0.5564]
# h750,g25,n=[100,200]:[0.8728,0.5610],[0.8722,0.5599]
# h1000,g25,n=[100,200]:[0.8725,0.5622],[0.8721,0.5637]
plt.plot([1,2,3], [0.8743,0.8747,0.8739] , label="h=500,g=25", marker="o", linewidth=linewidth,ms=ms)
plt.plot([2,3], [0.8744,0.8750] , label="h=500,g=50", marker="v", linewidth=linewidth,ms=ms)
plt.plot([1,3], [0.8728,0.8722] , label="h=750,g=25", marker="s", linewidth=linewidth,ms=ms)
plt.plot([1,3], [0.8725,0.8721] , label="h=1000,g=25", marker="*", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)


plt.subplot(8,3,9) # rmse
plt.title("with G, w/o A", fontsize=fontsize, weight='bold')
plt.xlabel("g", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[25, 50, 100], fontsize=fontsize_axis)
plt.ylabel("RMSE", fontsize=fontsize_axis)
plt.ylim(0.871, 0.879)
plt.yticks([0.871, 0.873, 0.875, 0.877, 0.879], fontsize=fontsize_axis)
plt.grid(axis='y')
# n150,h500,g=[25,50]:[0.8747,0.5624],[0.8744,0.5600]
# n200,h500,g=[25,50]:[0.8739,0.5617],[0.8750,0.5564]
# n200,h750,g=[25,50,100]:[0.8722,0.5599],[0.8738,0.5639],[0.8756,0.5614]
# n200,h1000,g=[25,50]:[0.8721,0.5637],[0.8735,0.5608]
plt.plot([1,2], [0.8747,0.8744] , label="n=150,h=500", marker="o", linewidth=linewidth,ms=ms)
plt.plot([1,2], [0.8739,0.8750] , label="n=200,h=500", marker="v", linewidth=linewidth,ms=ms)
plt.plot([1,2,3], [0.8722,0.8738,0.8756] , label="n=200,h=750", marker="s", linewidth=linewidth,ms=ms)
plt.plot([1,2], [0.8721,0.8735] , label="n=200,h=1000", marker="*", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)


plt.subplot(8,3,10) # recall
plt.xlabel("h", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[500, 750, 1000], fontsize=fontsize_axis)
plt.ylabel("Recall", fontsize=fontsize_axis)
plt.ylim(0.55, 0.566)
plt.yticks([0.55, 0.554, 0.558, 0.562, 0.566], fontsize=fontsize_axis)
plt.grid(axis='y')
# n100,g25,h=[500, 750, 1000]:[0.8743,0.5572],[0.8728,0.5610],[0.8725,0.5622]
# n200,g25,h=[500, 750, 1000]:[0.8739,0.5617],[0.8722,0.5599],[0.8721,0.5637]
# n200,g50,h=[500, 750, 1000]:[0.8750,0.5564],[0.8738,0.5639],[0.8735,0.5608]
plt.plot([1,2,3], [0.5572,0.5610,0.5622] , label="n=150,g=25", marker="o", linewidth=linewidth,ms=ms)
plt.plot([1,2,3], [0.5617,0.5599,0.5637] , label="n=200,g=25", marker="v", linewidth=linewidth,ms=ms)
plt.plot([1,2,3], [0.5564,0.5639,0.5608] , label="n=200,g=50", marker="s", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)


plt.subplot(8,3,11) # recall
plt.xlabel("n", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[100, 150, 200], fontsize=fontsize_axis)
plt.ylabel("Recall", fontsize=fontsize_axis)
plt.ylim(0.55, 0.566)
plt.yticks([0.55, 0.554, 0.558, 0.562, 0.566], fontsize=fontsize_axis)
plt.grid(axis='y')
# h500,g25,n=[100,150,200]:[0.8743,0.5572],[0.8747,0.5624],[0.8739,0.5617]
# h500,g50,n=[150,200]:[0.8744,0.5600],[0.8750,0.5564]
# h750,g25,n=[100,200]:[0.8728,0.5610],[0.8722,0.5599]
# h1000,g25,n=[100,200]:[0.8725,0.5622],[0.8721,0.5637]
plt.plot([1,2,3], [0.5572,0.5624,0.5617] , label="h=500,g=25", marker="o", linewidth=linewidth,ms=ms)
plt.plot([2,3], [0.5600,0.5564] , label="h=500,g=50", marker="v", linewidth=linewidth,ms=ms)
plt.plot([1,3], [0.5610,0.5599] , label="h=750,g=25", marker="s", linewidth=linewidth,ms=ms)
plt.plot([1,3], [0.5622,0.5637] , label="h=1000,g=25", marker="*", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)


plt.subplot(8,3,12) # recall
plt.xlabel("g", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[25, 50, 100], fontsize=fontsize_axis)
plt.ylabel("Recall", fontsize=fontsize_axis)
plt.ylim(0.55, 0.566)
plt.yticks([0.55, 0.554, 0.558, 0.562, 0.566], fontsize=fontsize_axis)
plt.grid(axis='y')
# n150,h500,g=[25,50]:[0.8747,0.5624],[0.8744,0.5600]
# n200,h500,g=[25,50]:[0.8739,0.5617],[0.8750,0.5564]
# n200,h750,g=[25,50,100]:[0.8722,0.5599],[0.8738,0.5639],[0.8756,0.5614]
# n200,h1000,g=[25,50]:[0.8721,0.5637],[0.8735,0.5608]
plt.plot([1,2], [0.5624,0.5600] , label="n=150,h=500", marker="o", linewidth=linewidth,ms=ms)
plt.plot([1,2], [0.5617,0.5564] , label="n=200,h=500", marker="v", linewidth=linewidth,ms=ms)
```

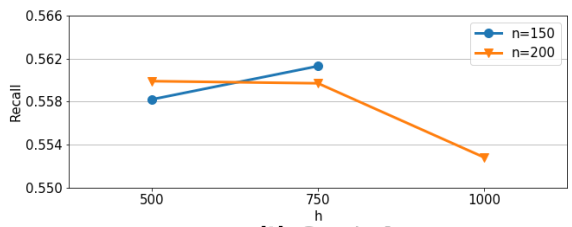
```
plt.plot([1,2,3], [0.5599,0.5639,0.5614] , label="n=200,h=750", marker="s", linewidth=linewidth,ms=ms)
plt.plot([1,2], [0.5637,0.5608] , label="n=200,h=1000", marker="*", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)


# ------------------------------------------------------------  w/o G, with A
plt.subplot(8,3,13) # rmse
plt.title("w/o G, with A", fontsize=fontsize, weight='bold')
plt.xlabel("h", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[500, 750, 1000], fontsize=fontsize_axis)
plt.ylabel("RMSE", fontsize=fontsize_axis)
plt.ylim(0.871, 0.879)
plt.yticks([0.871, 0.873, 0.875, 0.877, 0.879], fontsize=fontsize_axis)
plt.grid(axis='y')
# n200,a25,h=[500,750,1000]: [0.8734,0.8730,0.8730],[0.5685,0.5613,0.5626]
# n200,a50,h=[500,750,1000]: [0.8749,0.8736,0.8726],[0.5643,0.5558,0.5608]
# n200,a100,h=[750,1000]: [0.8732,0.8782],[0.5635,0.5617]
plt.plot([1,2,3], [0.8734,0.8730,0.8730] , label="n=200,a=25", marker="o", linewidth=linewidth,ms=ms)
plt.plot([1,2,3], [0.8749,0.8736,0.8726] , label="n=200,a=50", marker="v", linewidth=linewidth,ms=ms)
plt.plot([2,3], [0.8732,0.8782] , label="n=200,a=100", marker="s", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)


plt.subplot(8,3,14) # rmse
plt.title("w/o G, with A", fontsize=fontsize, weight='bold')
plt.xlabel("n", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[100,150,200], fontsize=fontsize_axis)
plt.ylabel("RMSE", fontsize=fontsize_axis)
plt.ylim(0.871, 0.879)
plt.yticks([0.871, 0.873, 0.875, 0.877, 0.879], fontsize=fontsize_axis)
plt.grid(axis='y')
# h500,a25,n=[150,200]: [0.8736,0.8734],[0.5586,0.5685]
# h500,a50,n=[150,200]: [0.8743,0.8749],[0.5625,0.5643]
# h1000,a15,n=[100,150,200]: [0.8730,0.8724,0.8733],[0.5547,0.5647,0.5622]
plt.plot([2,3], [0.8736,0.8734] , label="h=500,a=25", marker="o", linewidth=linewidth,ms=ms)
plt.plot([2,3], [0.8743,0.8749] , label="h=500,a=50", marker="v", linewidth=linewidth,ms=ms)
plt.plot([1,2,3], [0.8730,0.8724,0.8733] , label="h=1000,a=15", marker="s", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)


plt.subplot(8,3,15) # rmse
plt.title("w/o G, with A", fontsize=fontsize, weight='bold')
plt.xlabel("a", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[25, 50, 100], fontsize=fontsize_axis)
plt.ylabel("RMSE", fontsize=fontsize_axis)
plt.ylim(0.871, 0.879)
plt.yticks([0.871, 0.873, 0.875, 0.877, 0.879], fontsize=fontsize_axis)
plt.grid(axis='y')
# n150,h500,a=[25,50]: [0.8736,0.8743], [0.5586,0.5625]
# n200,h500,a=[25,50]: [0.8734,0.8749],[0.5685,0.5643]
# n200,h750,a=[25,50,100]: [0.8730,0.8736,0.8732],[0.5613,0.5558,0.5635]
# n200,h1000,a=[25,50,100]: [0.8730,0.8726,0.8782],[0.5626,0.5608,0.5617]
plt.plot([1,2], [0.8736,0.8743] , label="n=150,h=500", marker="o", linewidth=linewidth,ms=ms)
plt.plot([1,2], [0.8734,0.8749] , label="n=200,h=500", marker="v", linewidth=linewidth,ms=ms)
plt.plot([1,2,3], [0.8730,0.8736,0.8732], label="n=200,h=750", marker="s", linewidth=linewidth,ms=ms)
plt.plot([1,2,3], [0.8730,0.8726,0.8782] , label="n=200,h=1000", marker="*", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)


plt.subplot(8,3,16) # recall
plt.xlabel("h", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[500, 750, 1000], fontsize=fontsize_axis)
plt.ylabel("Recall", fontsize=fontsize_axis)
plt.ylim(0.554, 0.578)
plt.yticks([0.554, 0.560, 0.566, 0.572, 0.578], fontsize=fontsize_axis)
plt.grid(axis='y')
# n200,a25,h=[500,750,1000]: [0.8734,0.8730,0.8730],[0.5685,0.5613,0.5626]
# n200,a50,h=[500,750,1000]: [0.8749,0.8736,0.8726],[0.5643,0.5558,0.5608]
# n200,a100,h=[750,1000]: [0.8732,0.8782],[0.5635,0.5617]
plt.plot([1,2,3], [0.5685,0.5613,0.5626] , label="n=200,a=25", marker="o", linewidth=linewidth,ms=ms)
plt.plot([1,2,3], [0.5643,0.5558,0.5608] , label="n=200,a=50", marker="v", linewidth=linewidth,ms=ms)
plt.plot([2,3], [0.5635,0.5617] , label="n=200,a=100", marker="s", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)

plt.subplot(8,3,17) # recall
plt.xlabel("n", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[100, 150, 200], fontsize=fontsize_axis)
plt.ylabel("Recall", fontsize=fontsize_axis)
plt.ylim(0.554, 0.578)
plt.yticks([0.554, 0.560, 0.566, 0.572, 0.578], fontsize=fontsize_axis)
plt.grid(axis='y')
# h500,a25,n=[150,200]: [0.8736,0.8734],[0.5586,0.5685]
# h500,a50,n=[150,200]: [0.8743,0.8749],[0.5625,0.5643]
# h1000,a15,n=[100,150,200]: [0.8730,0.8724,0.8733],[0.5547,0.5647,0.5622]
```

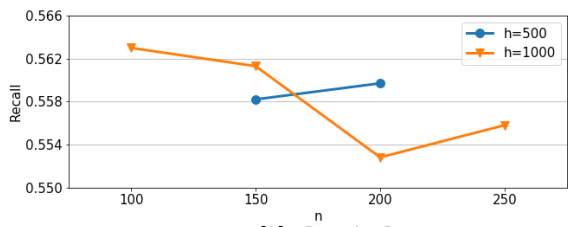
```
plt.plot([2,3], [0.5586,0.5685] , label="h=500,a=25", marker="o", linewidth=linewidth,ms=ms)
plt.plot([2,3], [0.5625,0.5643] , label="h=500,a=50", marker="v", linewidth=linewidth,ms=ms)
plt.plot([1,2,3], [0.5547,0.5647,0.5622], label="h=1000,a=15", marker="s", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)


plt.subplot(8,3,18) # recall
plt.xlabel("g", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[25, 50, 100], fontsize=fontsize_axis)
plt.ylabel("Recall", fontsize=fontsize_axis)
plt.ylim(0.554, 0.578)
plt.yticks([0.554, 0.560, 0.566, 0.572, 0.578], fontsize=fontsize_axis)
plt.grid(axis='y')
# n150,h500,a=[25,50]: [0.8736,0.8743], [0.5586,0.5625]
# n200,h500,a=[25,50]: [0.8734,0.8749],[0.5685,0.5643]
# n200,h750,a=[25,50,100]: [0.8730,0.8736,0.8732],[0.5613,0.5558,0.5635]
# n200,h1000,a=[25,50,100]: [0.8730,0.8726,0.8782],[0.5626,0.5608,0.5617]
plt.plot([1,2], [0.5586,0.5625] , label="n=150,h=500", marker="o", linewidth=linewidth,ms=ms)
plt.plot([1,2], [0.5685,0.5643] , label="n=200,h=500", marker="v", linewidth=linewidth,ms=ms)
plt.plot([1,2,3], [0.5613,0.5558,0.5635], label="n=200,h=750", marker="s", linewidth=linewidth,ms=ms)
plt.plot([1,2,3], [0.5626,0.5608,0.5617] , label="n=200,h=1000", marker="*", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)


# ------------------------------------------------------------ with G, with A
plt.subplot(8,3,19) # rmse
plt.title("with G, with A", fontsize=fontsize, weight='bold')
plt.xlabel("g", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[10, 15, 25], fontsize=fontsize_axis)
plt.ylabel("RMSE", fontsize=fontsize_axis)
plt.ylim(0.867, 0.871)
plt.yticks([0.867,0.868,0.869,0.870,0.871], fontsize=fontsize_axis)
plt.grid(axis='y')
# a10, g=[10,15,25]: [0.8688,0.8684,0.8689],[0.5698,0.5676,0.5695]
# a15, g=[10,15,25]: [0.8690,0.8691,0.8685],[0.5666,0.5653,0.5648]
# a25, g=[10,15,25]: [0.8682,0.8688,0.8699],[0.5667,0.5682,0.5640]
plt.plot([1,2,3],[0.8688,0.8684,0.8689] , label="a=10", marker="o", linewidth=linewidth,ms=ms)
plt.plot([1,2,3],[0.8690,0.8691,0.8685] , label="a=15", marker="v", linewidth=linewidth,ms=ms)
plt.plot([1,2,3],[0.8682,0.8688,0.8699] , label="a=25", marker="s", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)


plt.subplot(8,3,20) # rmse
plt.title("with G, with A", fontsize=fontsize, weight='bold')
plt.xlabel("a", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[10,15,25], fontsize=fontsize_axis)
plt.ylabel("RMSE", fontsize=fontsize_axis)
plt.ylim(0.867, 0.871)
plt.yticks([0.867,0.868,0.869,0.870,0.871], fontsize=fontsize_axis)
plt.grid(axis='y')
# g10, a=[10,15,25]: [0.8688,0.8690,0.8682],[0.5698,0.5666,0.5667]
# g15, a=[10,15,25]: [0.8684,0.8691,0.8688], [0.5676,0.5653,0.5682]
# g25, a=[10,15,25]: [0.8689,0.8685,0.8699], [0.5695,0.5648,0.5640]
plt.plot([1,2,3],[0.8688,0.8690,0.8682] , label="g=10", marker="o", linewidth=linewidth,ms=ms)
plt.plot([1,2,3],[0.8684,0.8691,0.8688] , label="g=15", marker="v", linewidth=linewidth,ms=ms)
plt.plot([1,2,3],[0.8689,0.8685,0.8699] , label="g=25", marker="s", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)


plt.subplot(8,3,22) # recall
plt.xlabel("g", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[10, 15, 25], fontsize=fontsize_axis)
plt.ylabel("RMSE", fontsize=fontsize_axis)
plt.ylim(0.563, 0.571)
plt.yticks([0.563, 0.565, 0.567, 0.569, 0.571], fontsize=fontsize_axis)
plt.grid(axis='y')
# a10, g=[10,15,25]: [0.8688,0.8684,0.8689],[0.5698,0.5676,0.5695]
# a15, g=[10,15,25]: [0.8690,0.8691,0.8685],[0.5666,0.5653,0.5648]
# a25, g=[10,15,25]: [0.8682,0.8688,0.8699],[0.5667,0.5682,0.5640]
plt.plot([1,2,3],[0.5698,0.5676,0.5695] , label="a=10", marker="o", linewidth=linewidth,ms=ms)
plt.plot([1,2,3],[0.5666,0.5653,0.5648] , label="a=15", marker="v", linewidth=linewidth,ms=ms)
plt.plot([1,2,3],[0.5667,0.5682,0.5640] , label="a=25", marker="s", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)


plt.subplot(8,3,23) # recall
plt.xlabel("a", fontsize=fontsize_axis)
plt.xlim(0.5,3.5)
plt.xticks([1,2,3],[10,15,25], fontsize=fontsize_axis)
plt.ylabel("RMSE", fontsize=fontsize_axis)
plt.ylim(0.563, 0.571)
plt.yticks([0.563, 0.565, 0.567, 0.569, 0.571], fontsize=fontsize_axis)
plt.grid(axis='y')
# g10, a=[10,15,25]: [0.8688,0.8690,0.8682],[0.5698,0.5666,0.5667]
# g15, a=[10,15,25]: [0.8684,0.8691,0.8688], [0.5676,0.5653,0.5682]
```

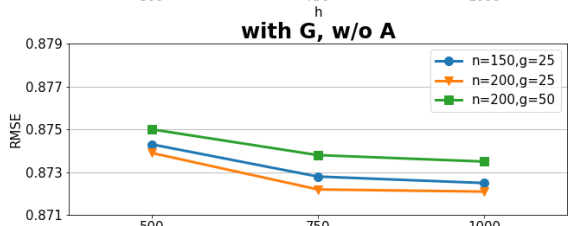
```
# g25, a=[10,15,25]: [0.8689,0.8685,0.8699], [0.5695,0.5648,0.5640]
plt.plot([1,2,3],[0.5698,0.5666,0.5667] , label="g=10", marker="o", linewidth=linewidth,ms=ms)
plt.plot([1,2,3],[0.5676,0.5653,0.5682] , label="g=15", marker="v", linewidth=linewidth,ms=ms)
plt.plot([1,2,3],[0.5695,0.5648,0.5640] , label="g=25", marker="s", linewidth=linewidth,ms=ms)
plt.legend(fontsize=fontsize_axis)

# plt.savefig("figs/ml-1m_DL-2.pdf")

plt.show()
```

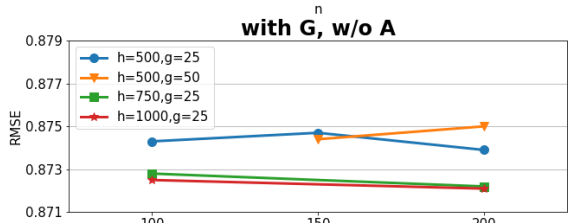


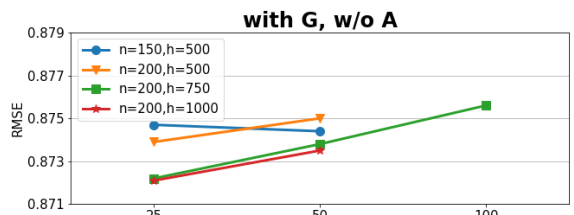


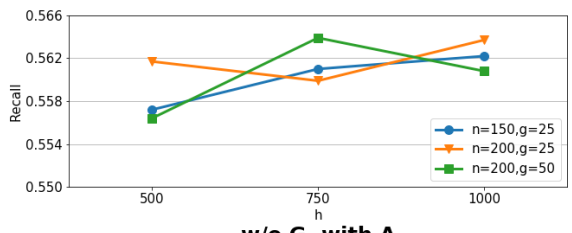


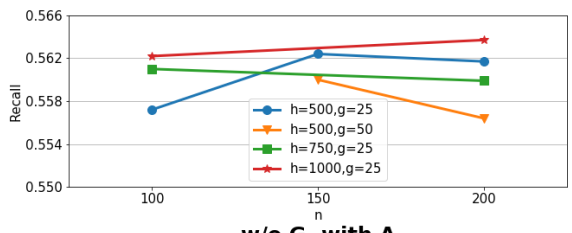


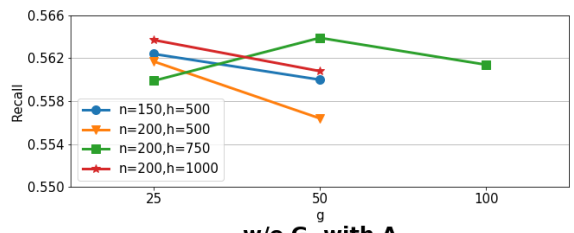


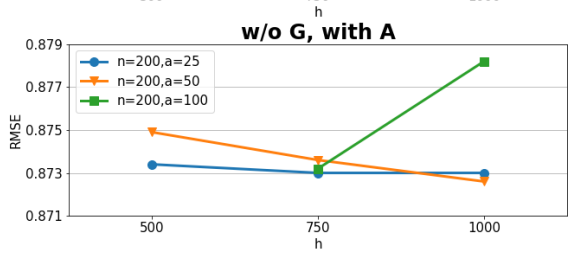


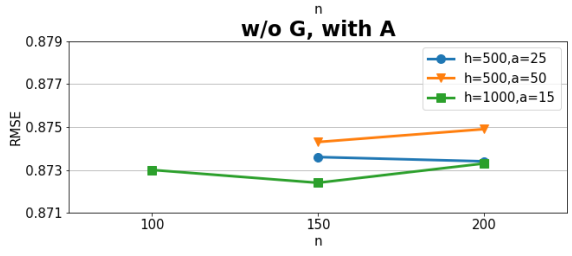


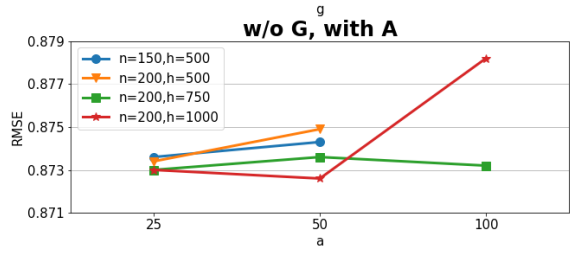


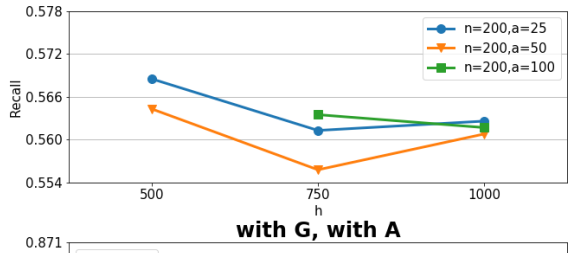


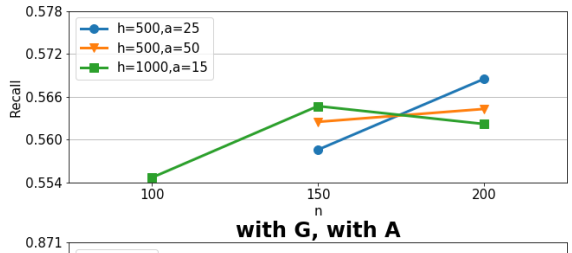


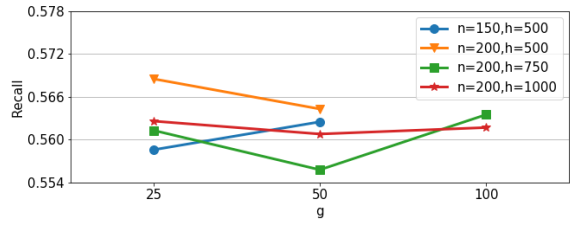


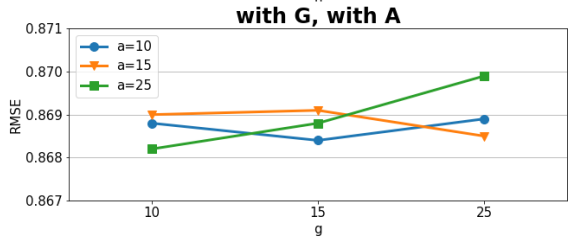


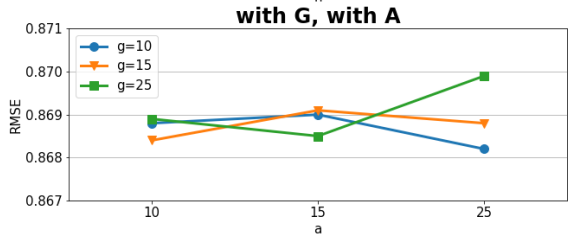


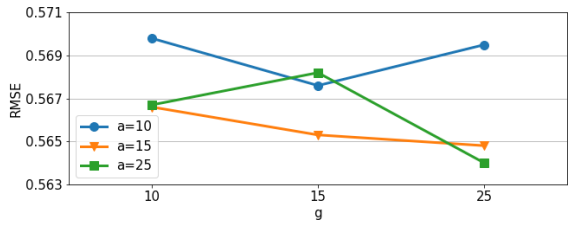


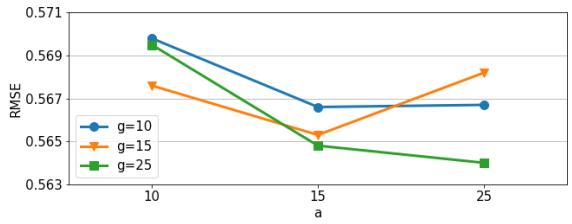












In [ ]: