Unsupervised Simulation under Bad distribution generation

Jiaxin Hu

8.29.2019

According to the *Paper Sketch of AISTATS*, when we choose bad distribution to generate the core tensor, the result would not be such good. In terms of MSE, we assume that constrains would improve the performance.

I ran simulations when the rank of core tensor is $(5, 5, 5)$ and the dimension of the tensor ranges from 20 to 70. Apply three constrains under unsupervised case: no constrain, vanilla, conjugate(penalty likelihood) constrain. The distributions of the core tensor entries are : $N(0, 1), N(10, 1), U(0, 1), U(0, 10)$.

1 MSE results

1.1 N(0,1)

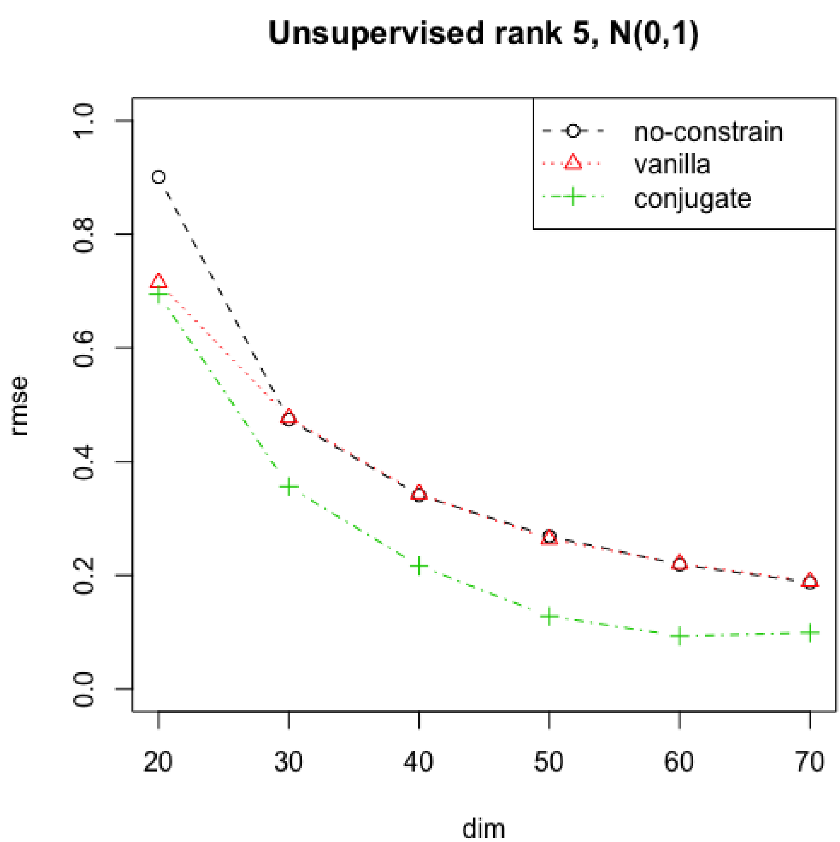

Figure 1: The core tensor entries are generated from $N(0, 1)$

Through *Figure 1*, under the case of $N(0,1)$ the adding constrains, especially conjugate(penalty likelihood) constrain can improves the performance of the MSE.

1.2 N(10,1)

As there are only partial result under the case $N(10,1), U(0,10)$, the *Figure 2,4* only show result when dimension is 20,40,60.

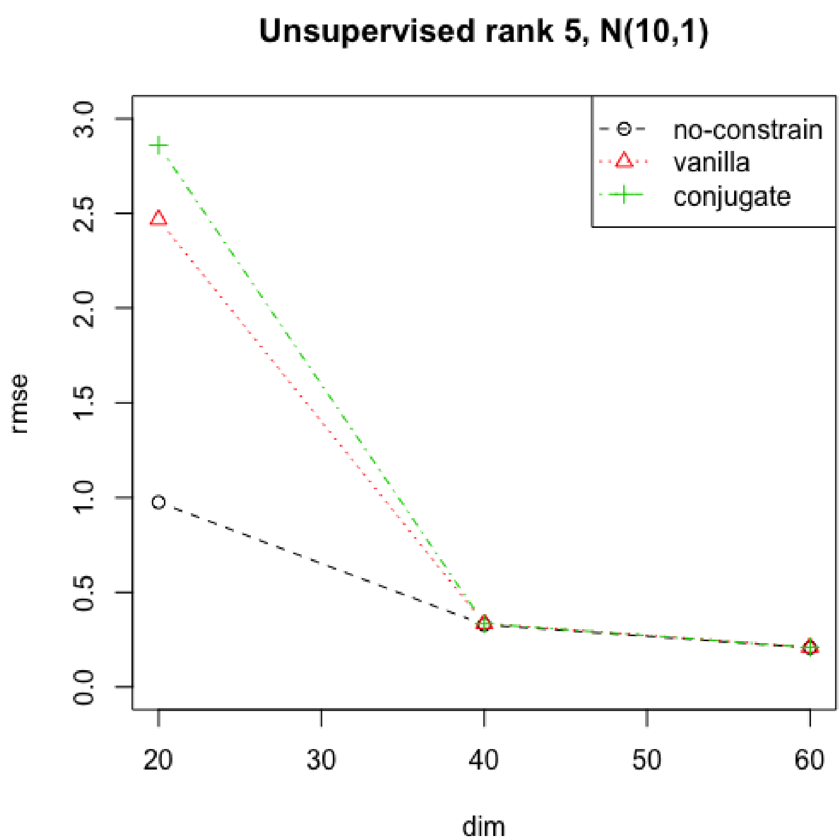

Figure 2: The core tensor entries are generated from $N(10,1)$

1.3 U(0,1)

Through *Figure 3*, under the case $U(0,1)$, adding conjugate and vanilla constrain can slightly improve the performance.

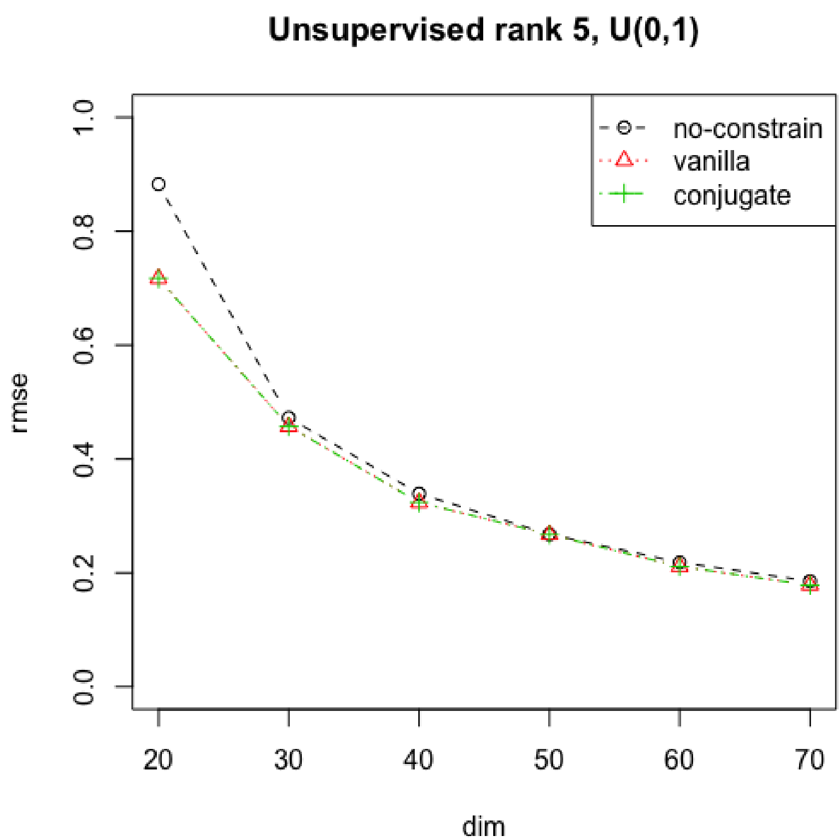

Figure 3: The core tensor entries are generated from $U(0,1)$

1.4 U(0,10)

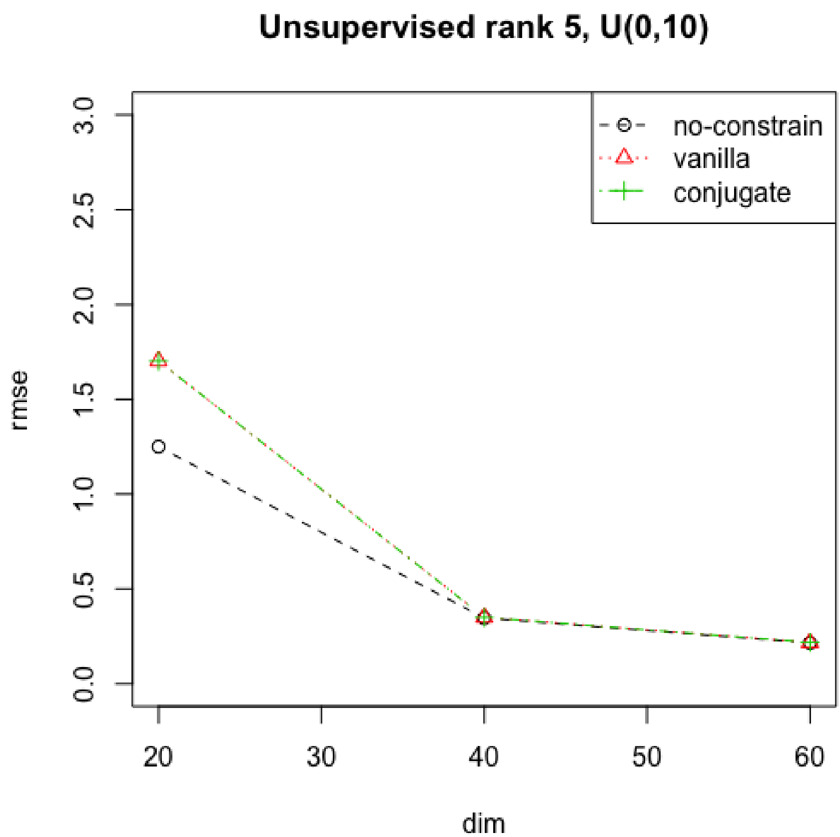

Figure 4: The core tensor entries are generated from $U(0, 10)$

2 True U vs Est U; True G vs Est G

This part shows the scatter plot of *True U vs Est U* and *True G vs Est G*, which can somehow display the influence from the constrains.
In this section, the setting is: rank of core tensor is 5, dimension of the tensor is 40.

2.1 True U vs Est U

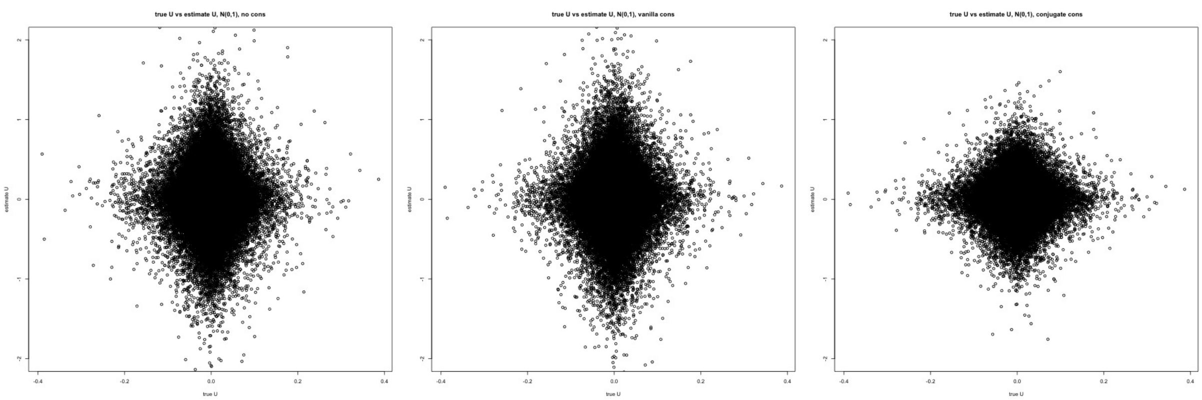

Figure 5: True U(x-axis) vs Est U(y-axis) when core tensor entries from $N(0,1)$; the magnitude of axis in three graphs are the same; from left to the right, the graph comes from no constrain/vanilla constrain/conjugate constrain case.

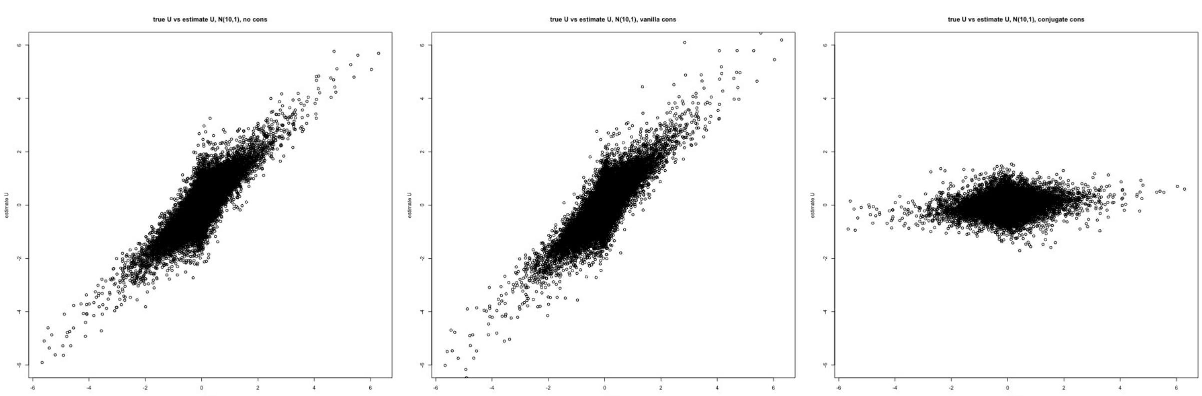

Figure 6: True U(x-axis) vs Est U(y-axis) when core tensor entries from $N(10,1)$; the magnitude of axis in three graphs are the same; from left to the right, the graph comes from no constrain/vanilla constrain/conjugate constrain case.

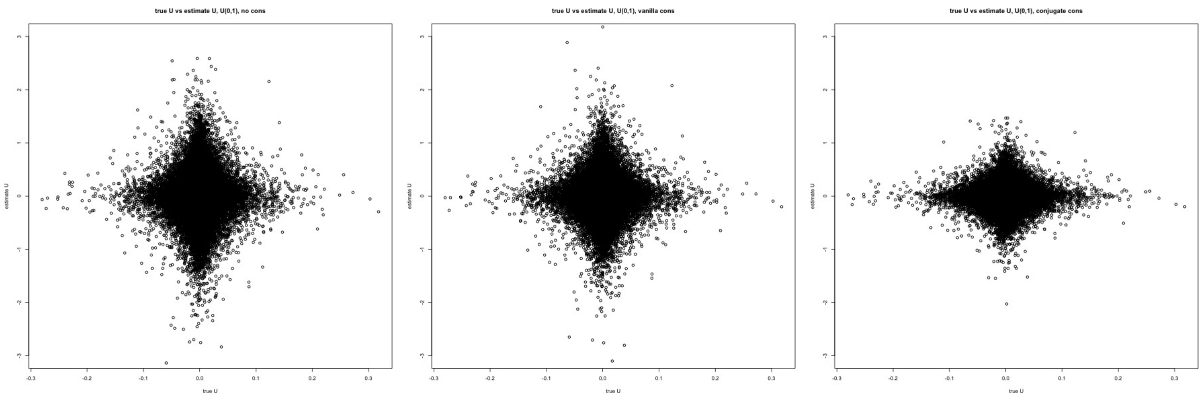

Figure 7: True U(x-axis) vs Est U(y-axis) when core tensor entries from $U(0, 1)$; the magnitude of axis in three graphs are the same; from left to the right, the graph comes from no constrain/vanilla constrain/conjugate constrain case.

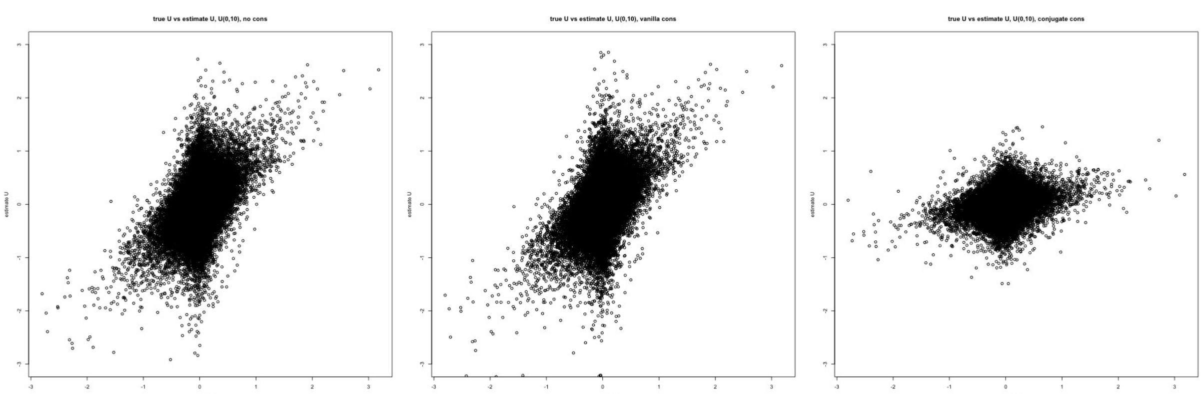

Figure 8: True U(x-axis) vs Est U(y-axis) when core tensor entries from $U(0, 10)$; the magnitude of axis in three graphs are the same; from left to the right, the graph comes from no constrain/vanilla constrain/conjugate constrain case.

2.2 True G vs Est G

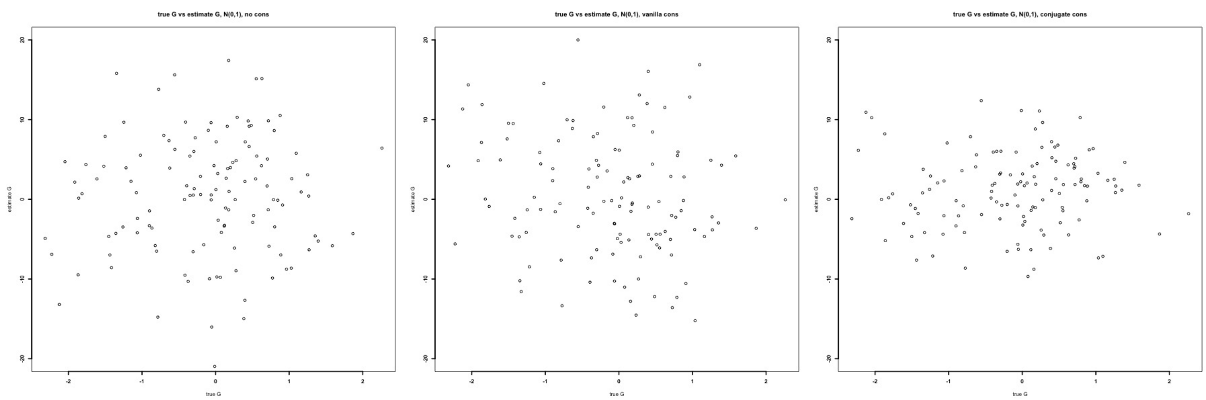

Figure 9: True G(x-axis) vs Est G(y-axis) when core tensor entries from $N(0,1)$; the magnitude of axis in three graphs are the same; from left to the right, the graph comes from no constrain/vanilla constrain/conjugate constrain case.

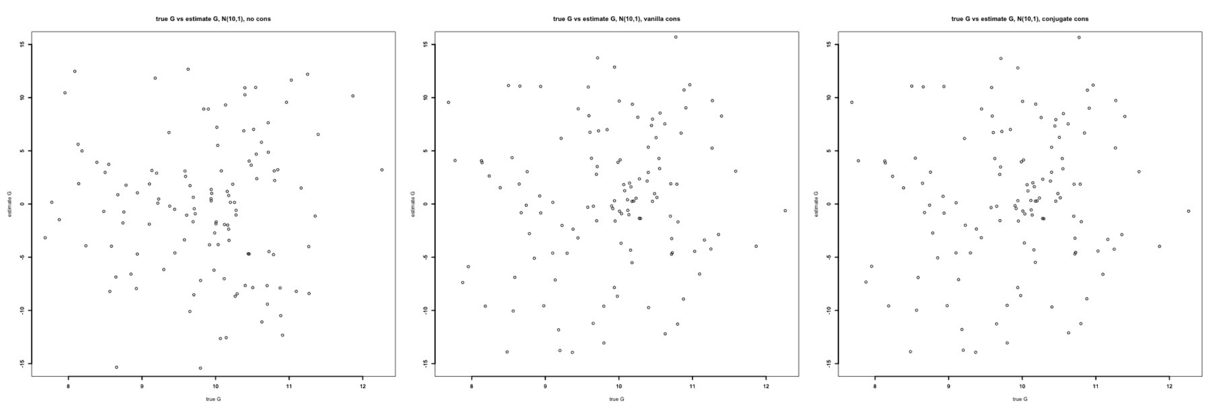

Figure 10: True G(x-axis) vs Est G(y-axis) when core tensor entries from $N(10,1)$; the magnitude of axis in three graphs are the same; from left to the right, the graph comes from no constrain/vanilla constrain/conjugate constrain case.

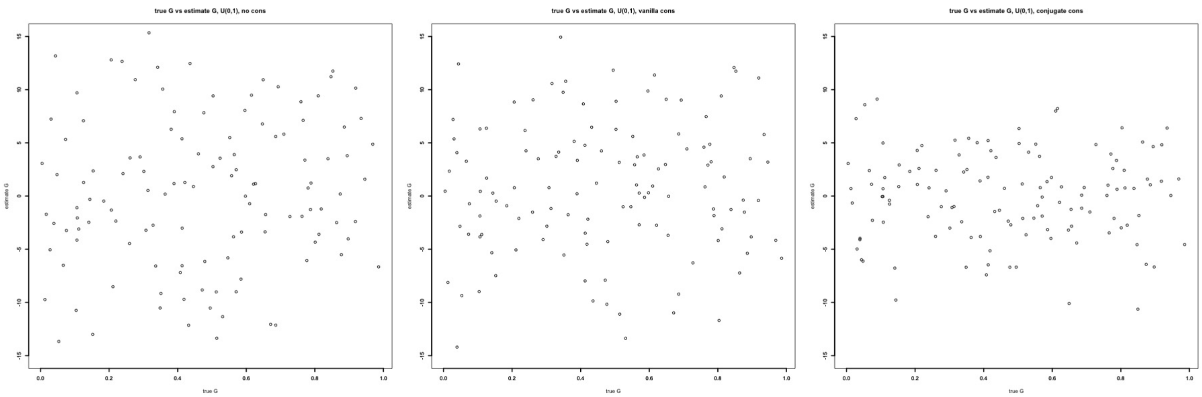

Figure 11: True G(x-axis) vs Est G(y-axis) when core tensor entries from $U(0,1)$; the magnitude of axis in three graphs are the same; from left to the right, the graph comes from no constrain/vanilla constrain/conjugate constrain case.

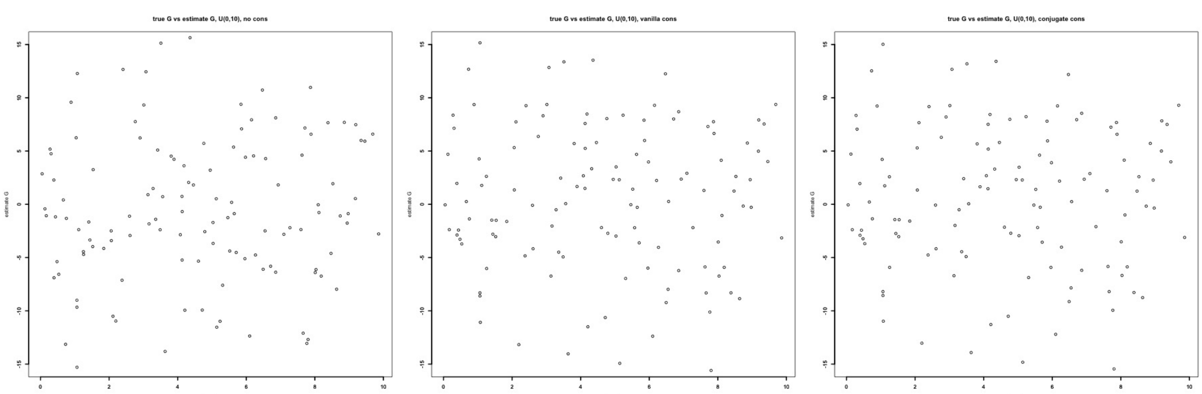

Figure 12: True G(x-axis) vs Est G(y-axis) when core tensor entries from $U(0,10)$; the magnitude of axis in three graphs are the same; from left to the right, the graph comes from no constrain/vanilla constrain/conjugate constrain case.

2.3 Analysis

In general, conjugate constrain have a relatively strong effect to the iteration process. Vanilla give a similar or slightly better performance than no constrains.

When the entries of U focus around the 0, like $N(0,1), U(0,1)$, it is difficult to estimate the true U and the pattern of true U vs U est would naturally be a "star shape". Under these cases, conjugate constrain can significantly decrease the magnitude of the entries of U and G (see in *Figure 5,7,9,11*), which leads to a better MSE performance. However, the conjugate constrain can not change the shape of pattern – still a "star" shape.

When the entries of U more scattered, like $N(10,1), U(0,10)$, the algorithm gives a fairly good result even without the constrain. And vanilla constrain doesn't effect much. However, conjugate constrain would shrink the U entries too much(see *Figure 6,8*). In the contrary, conjugate constrain gives the similar result on estimate G (see *Figure 10,12*) and give a similar MSE performance with other two cases when $dim = 40$.