Singular Value Decomposition for prevention of the Graph Convolutional Network overfitting

Anastasia Remizova
Ildar Abdrakhmanov
Mikhail Salnikov

Skoltech

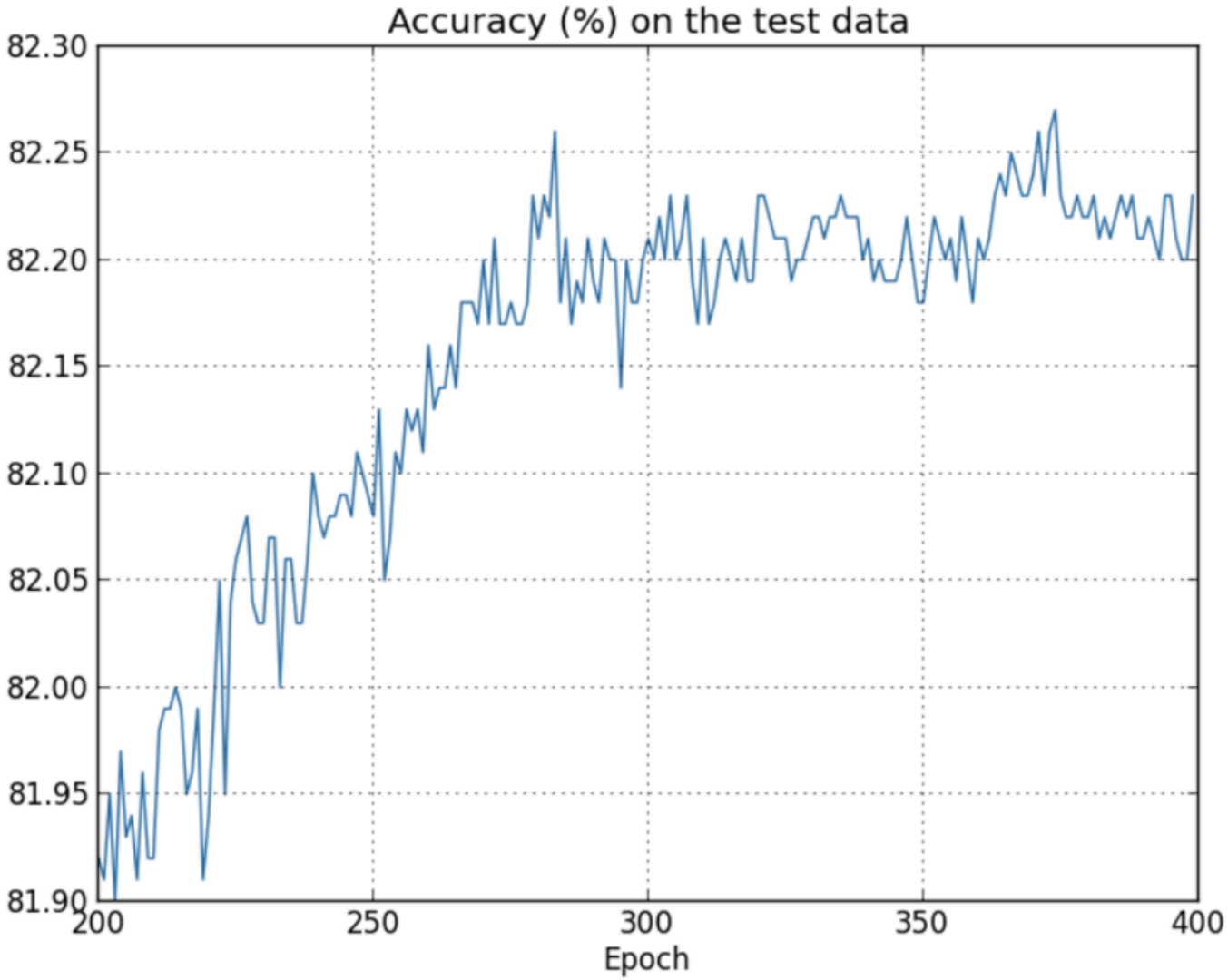
Problem Statement

Overfitting

- Close fit to train set
- Huge amount of neurons and layers
- Inability to make predictions

Common techniques:

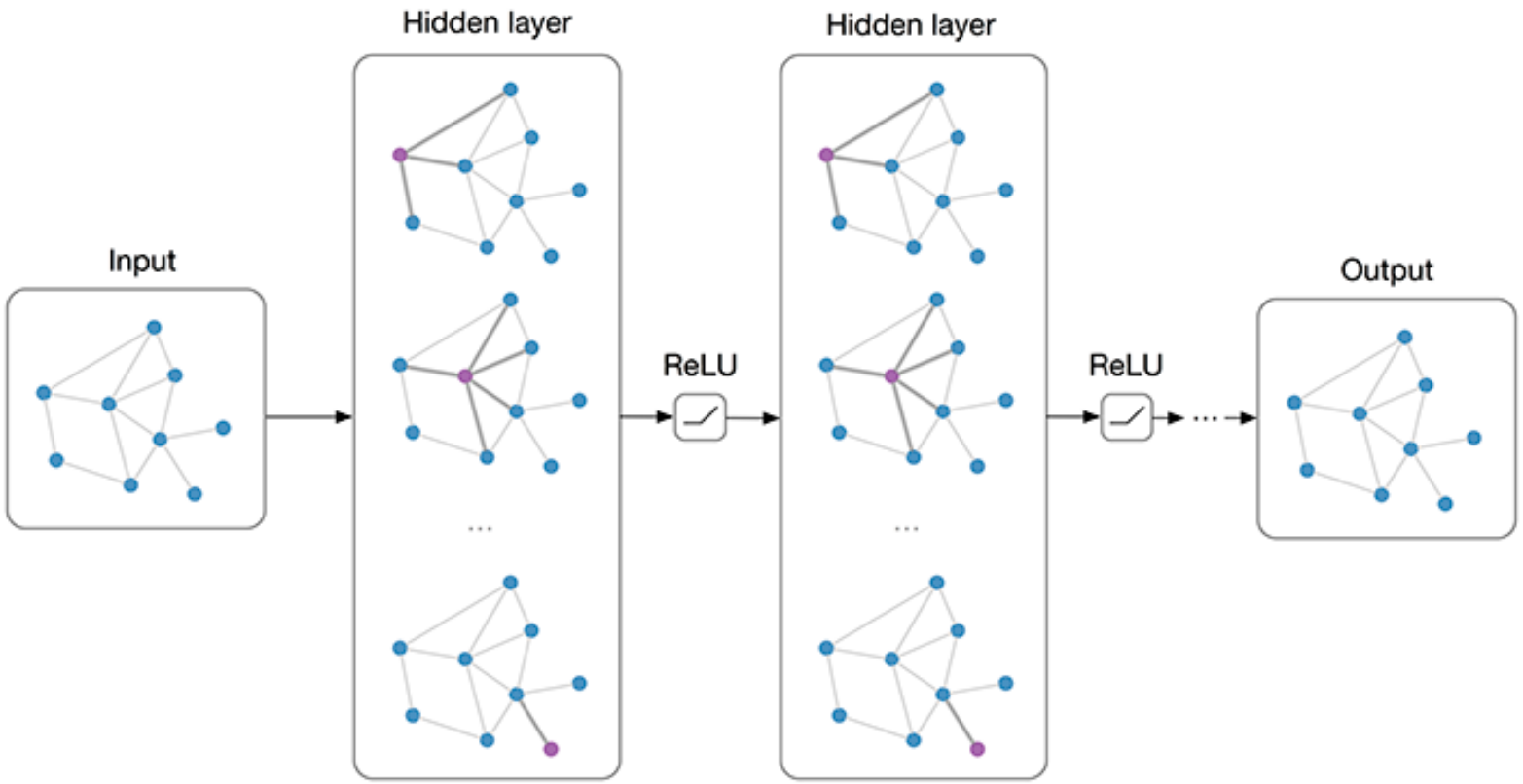
- Adding more data
- Data augmentation
- Complexity reduction
- Dropout



Graph Convolutional Network

Graph Neural Net

- Social networks
- Knowledge graphs
- Protein-interaction networks
- The World Wide Web

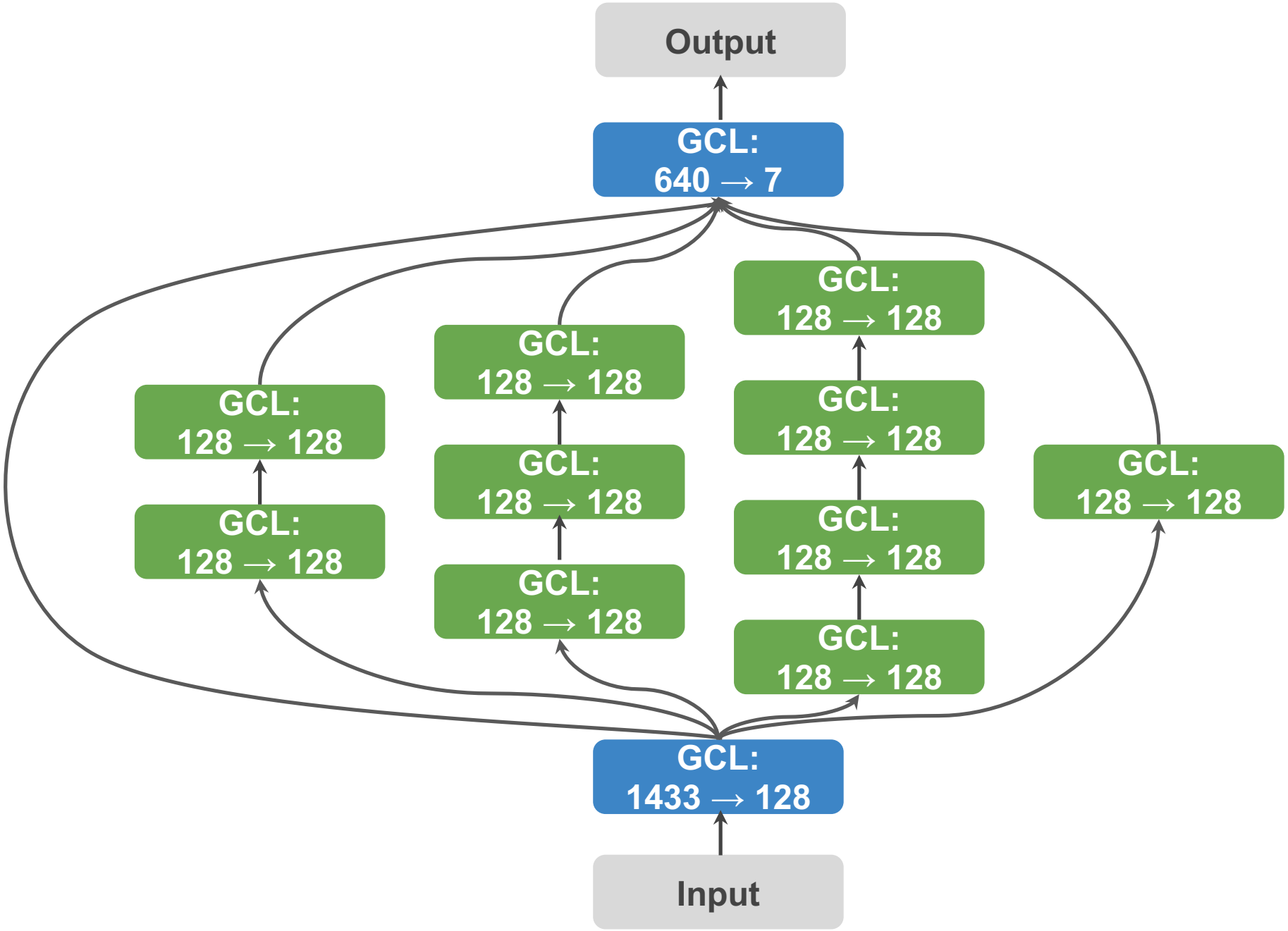


Applications

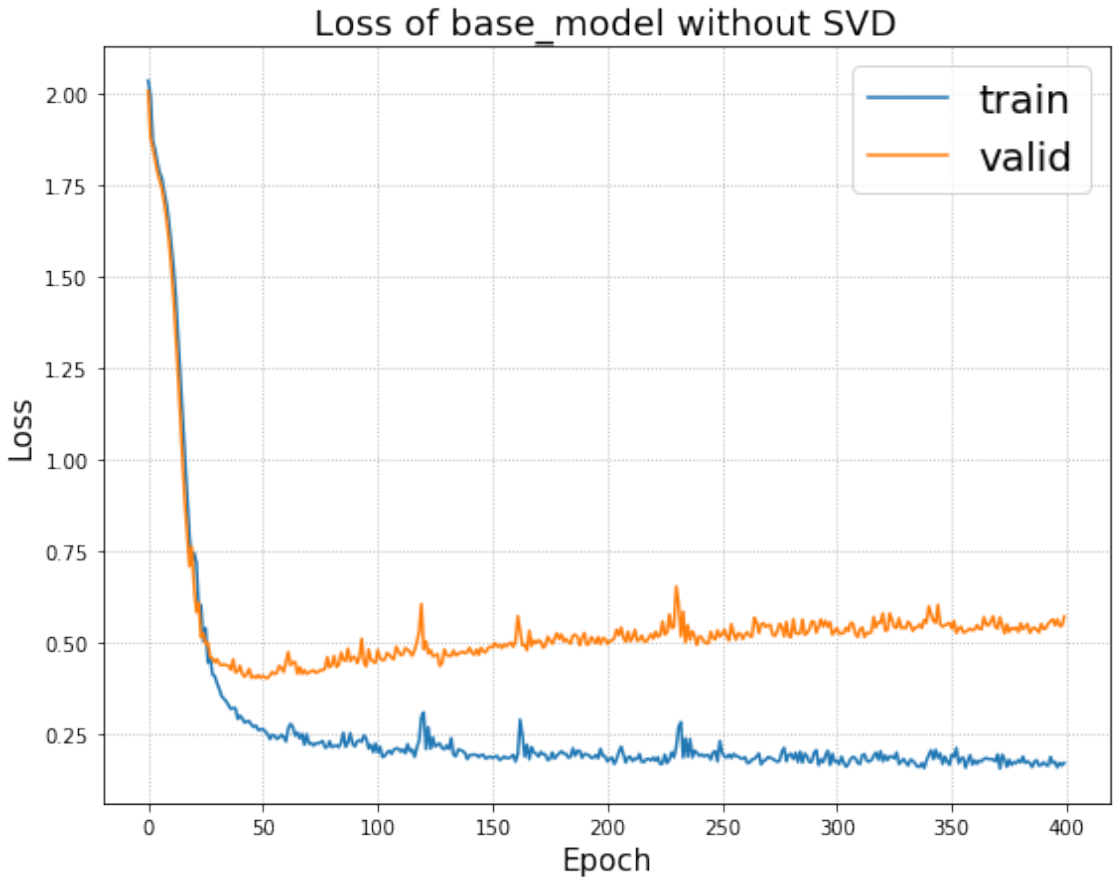
- Node classification
- Social recommendation
- Link prediction

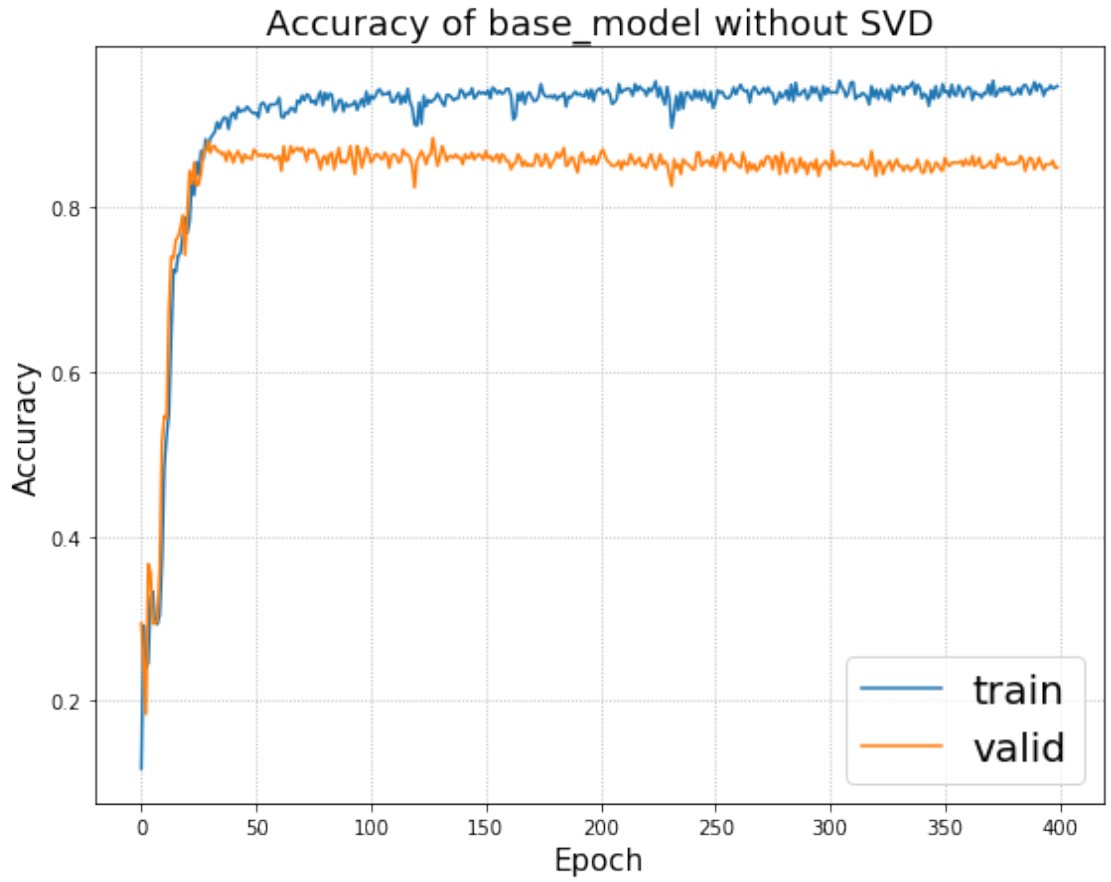
Why?

- Typically shallow
- Overfitting - main issue in deep GCN
- Dropout and weight penalizing don’t help


Output
GCL:
640 → 7
GCL:
128 → 128
GCL:
128 → 128
GCL:
128 → 128
GCL:
128 → 128
GCL:
128 → 128
GCL:
128 → 128
GCL:
128 → 128
GCL:
128 → 128
GCL:
1433 → 128
Input

Overfitting



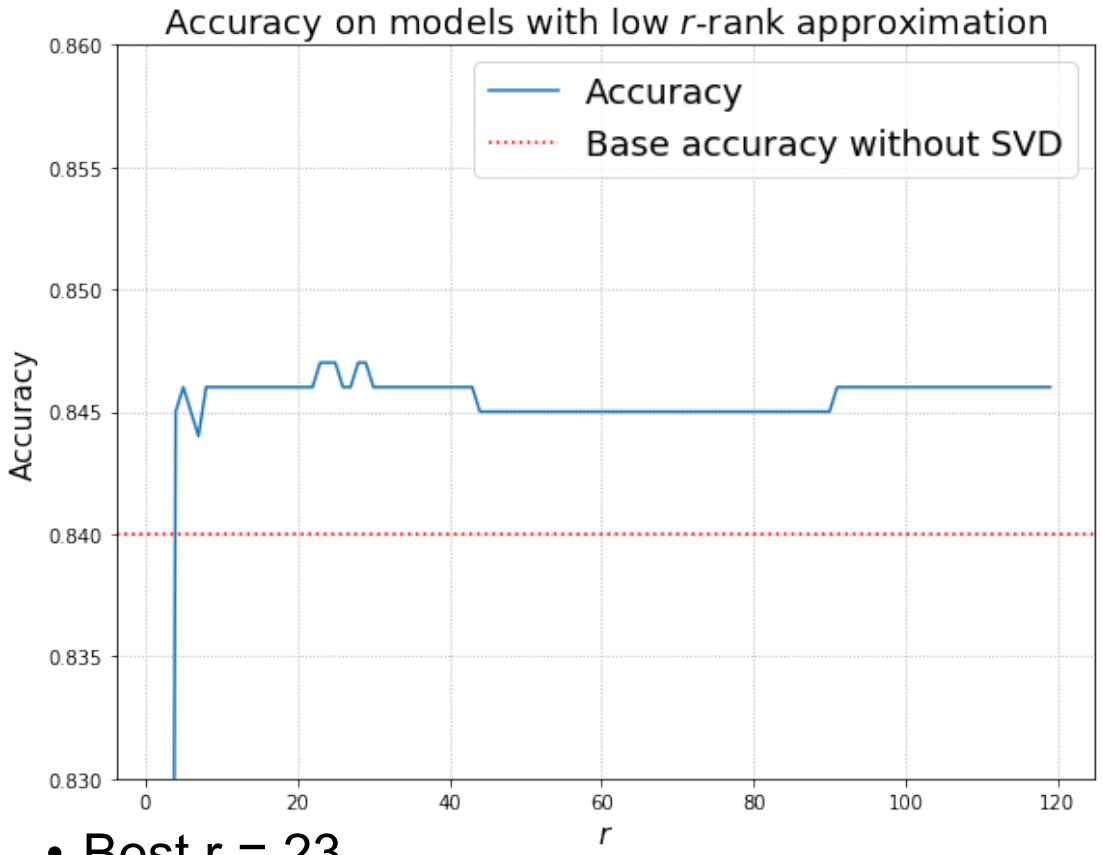


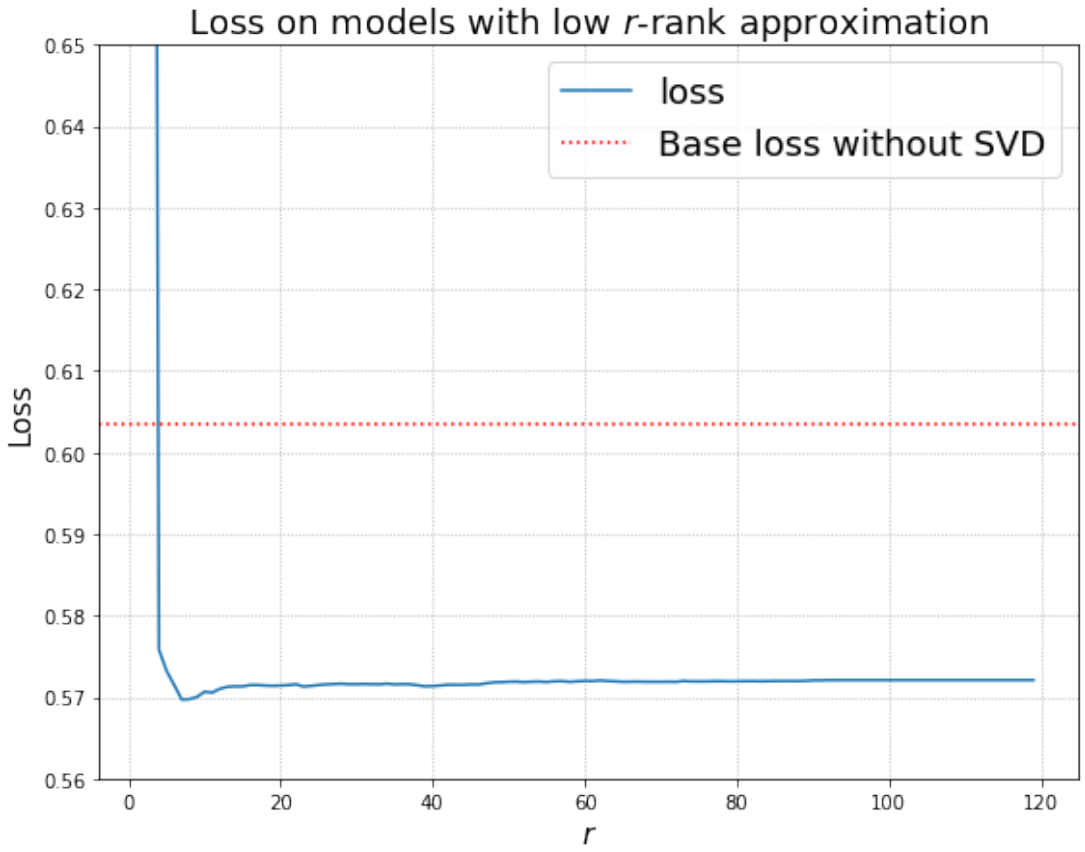
- Beyond 50 epoch model overfits
- Loss: 0.6
- Accuracy: 0.84

SVD vs Overfitting

- Recap SVD on fully-connected layers: $A = USV^T \rightarrow A_r = U_r S_r V_r^T$

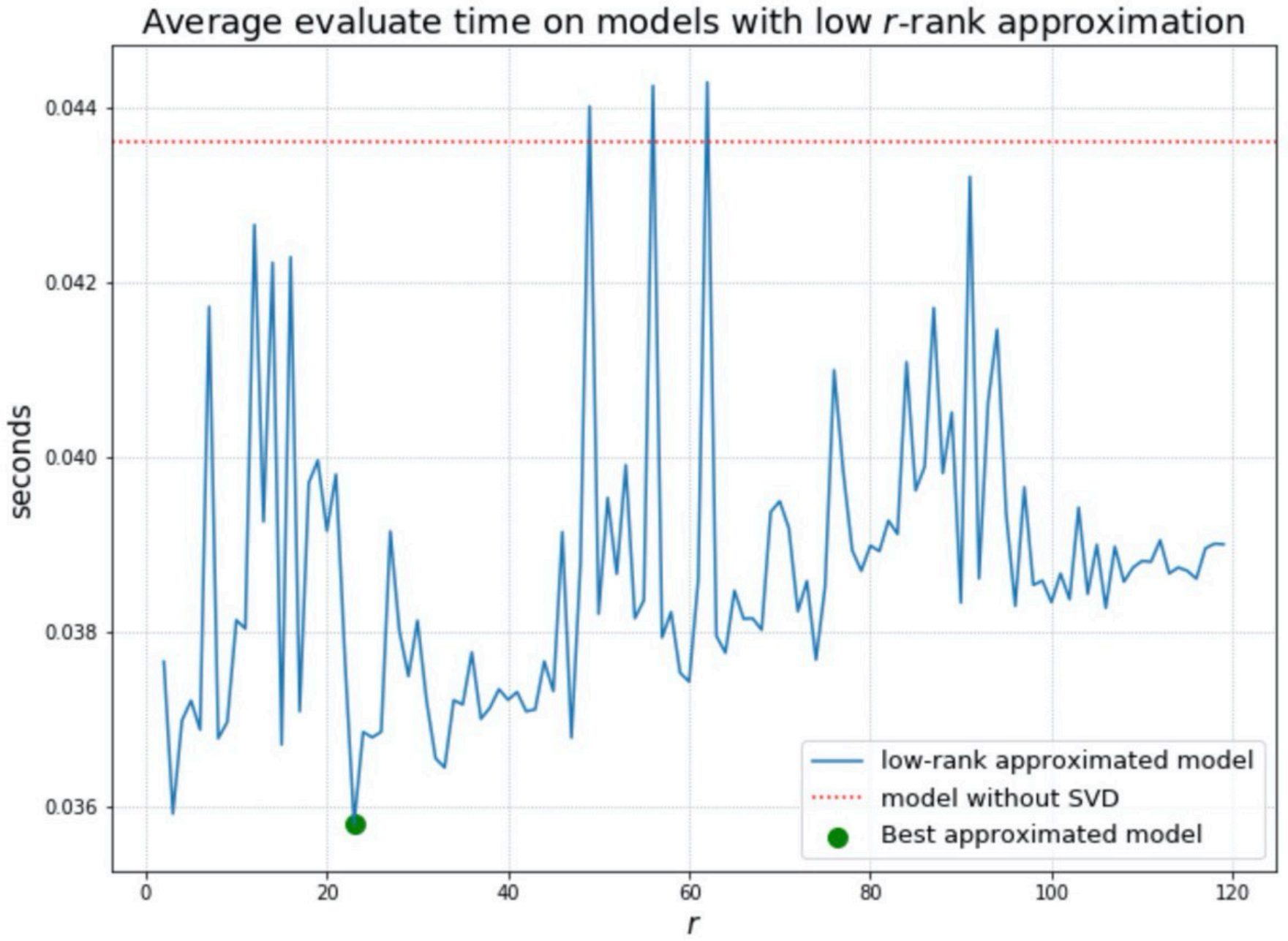
$$Ax + b \rightarrow (U_r S_r V_r^T)x + b = U_r(S_r V_r^T x) + b$$





- Best r = 23
- Accuracy: 0.85; Loss: 0.57

Bonus


Average evaluate time on models with low r-rank approximation
seconds
0.044
0.042
0.040
0.038
0.036
0
20
40
60
80
100
120
r
low-rank approximated model
model without SVD
Best approximated model

Thank you for your attention !