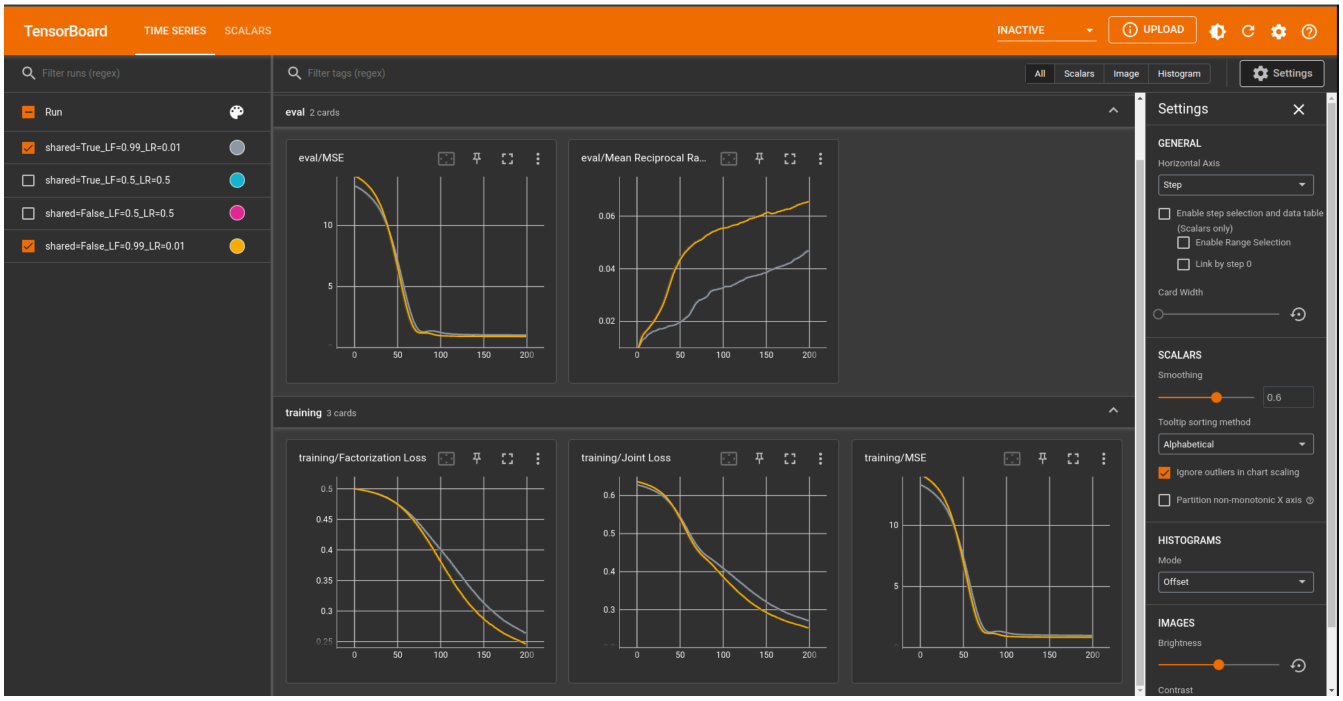
This handout includes space for every question that requires a written response. Please feel free to use it to handwrite your solutions (legibly, please). If you choose to typeset your solutions, the `README.md` for this assignment includes instructions to regenerate this handout with your typeset LaTeX solutions.

3.a



Run
- [✓] shared=True_LF=0.99_LR=0.01 ●
- [ ] shared=True_LF=0.5_LR=0.5 ●
- [ ] shared=False_LF=0.5_LR=0.5 ●
- [✓] shared=False_LF=0.99_LR=0.01 ●

Settings

GENERAL
Horizontal Axis: Step
- [ ] Enable step selection and data table (Scalars only)
- [ ] Enable Range Selection
- [ ] Link by step 0

Card Width

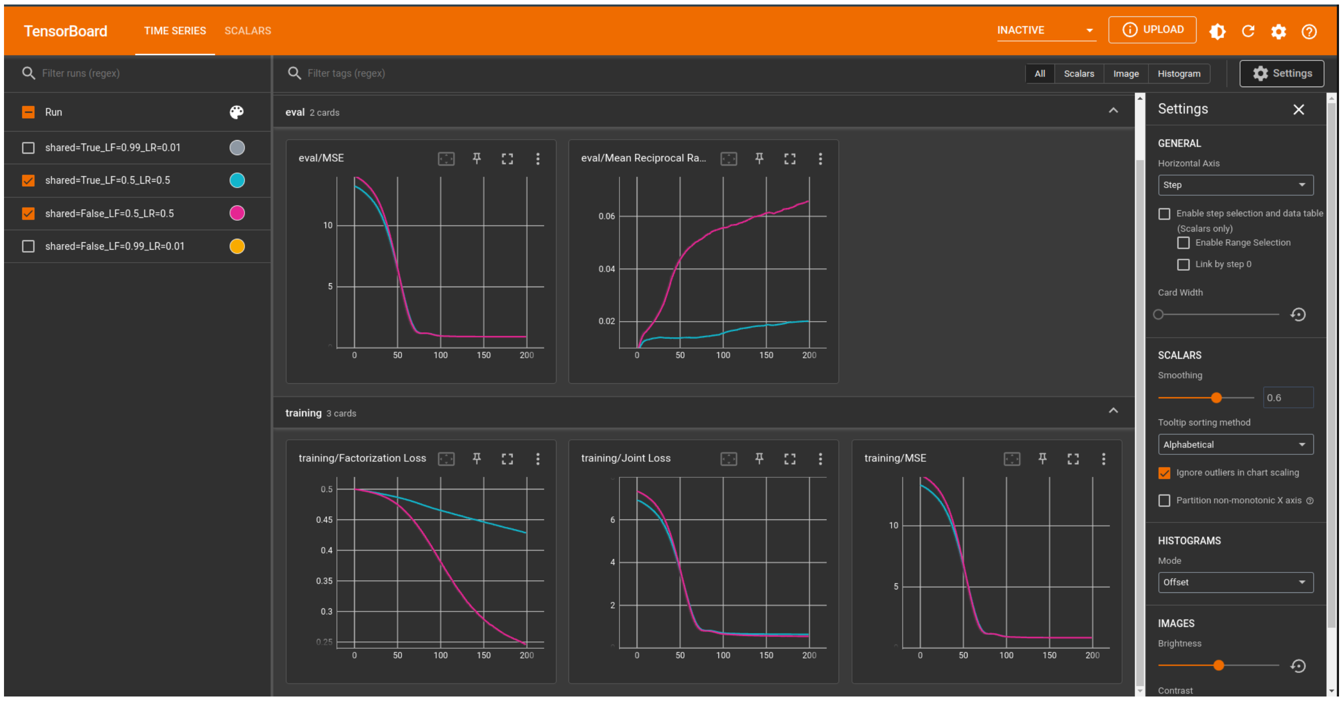
SCALARS
Smoothing: 0.6
Tooltip sorting method: Alphabetical
- [✓] Ignore outliers in chart scaling
- [ ] Partition non-monotonic X axis

HISTOGRAMS
Mode: Offset

IMAGES
Brightness
Contrast

Based on the train/test loss curves, parameter sharing does not outperform having separate models.
Looking at the curves we see that the separate models model has a similar MSE for both eval and training implying that both parameter sharing and separate models performed equally on the regression task.
The factorization and joint loss are lower and the Mean Reciprocal Rank is higher for the separate models model, impyling that the separate models model was more likely to recommend movies the user has seen than the shared parameters model.

3.b



Run
- [ ] shared=True_LF=0.99_LR=0.01
- [x] shared=True_LF=0.5_LR=0.5
- [x] shared=False_LF=0.5_LR=0.5
- [ ] shared=False_LF=0.99_LR=0.01

Settings

GENERAL
Horizontal Axis: Step
- [ ] Enable step selection and data table (Scalars only)
  - [ ] Enable Range Selection
  - [ ] Link by step 0

Card Width

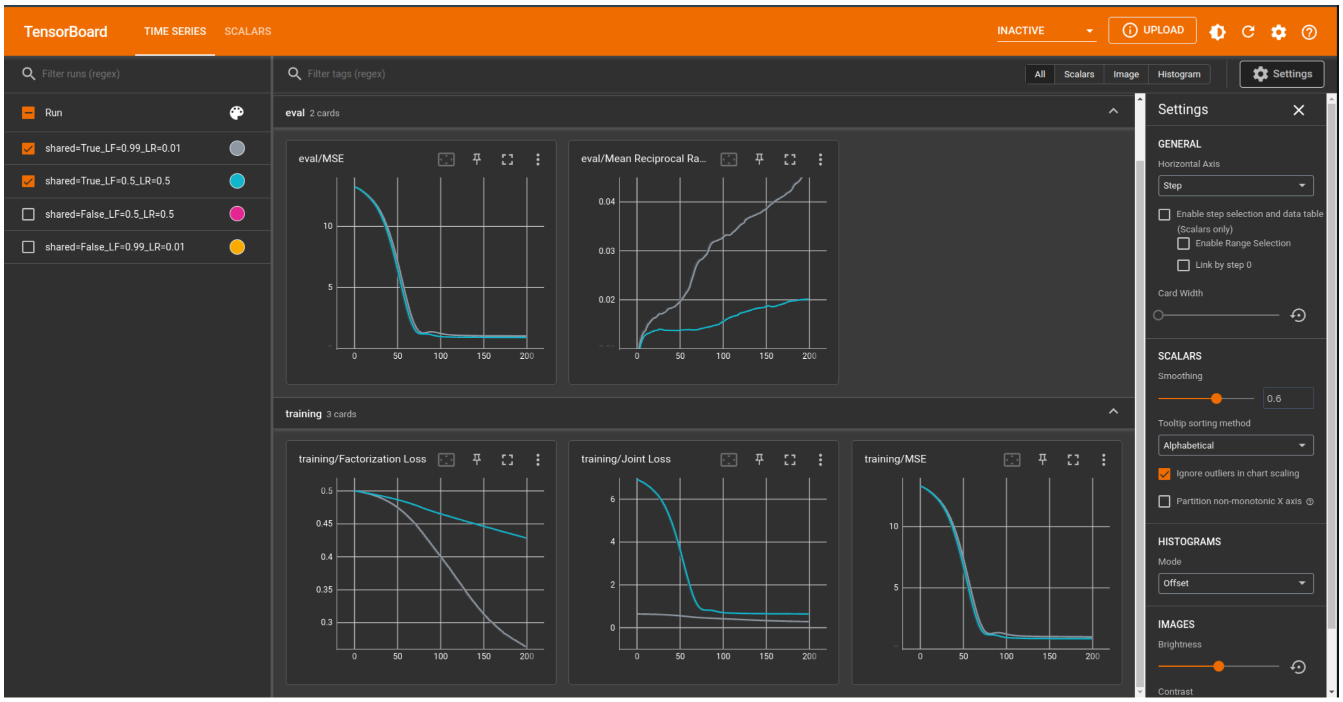
SCALARS
Smoothing: 0.6
Tooltip sorting method: Alphabetical
- [x] Ignore outliers in chart scaling
- [ ] Partition non-monotonic X axis

HISTOGRAMS
Mode: Offset

IMAGES
Brightness
Contrast

Based on the train/test loss curves, parameter sharing does not outperform having separate models once again.
First looking at the Joint Loss and MSE, parameter sharing and separate models look to produce similar results.
However, when looking at Factorization loss we see that separate models is significantly lower.
And when looking at the Mean Reciprical Rank we see once again that separate models is higher.

3.c



Run
- [✓] shared=True_LF=0.99_LR=0.01
- [✓] shared=True_LF=0.5_LR=0.5
- [ ] shared=False_LF=0.5_LR=0.5
- [ ] shared=False_LF=0.99_LR=0.01

Settings

GENERAL
Horizontal Axis: Step
- [ ] Enable step selection and data table (Scalars only)
- [ ] Enable Range Selection
- [ ] Link by step 0

Card Width

SCALARS
Smoothing: 0.6
Tooltip sorting method: Alphabetical
- [✓] Ignore outliers in chart scaling
- [ ] Partition non-monotonic X axis

HISTOGRAMS
Mode: Offset

IMAGES
Brightness
Contrast

Comparing the two shared models we see that they have similar performance on MSE loss.
However, when comparing the Factorization and Joint Loss as well as the Mean Reciprocal Rank we see that the model with $\lambda_F = 0.99$ $\lambda_R = 0.01$ converges faster.
This performance difference makes sense as the model with $\lambda_F = 0.99$ $\lambda_R = 0.01$ is weighted in favor of the factorization task.
Why does weighing the model in favor of the factorization task generate better results?
I'm not entirely certain, but my best guess / intuition is that the factorization task is a more complicated task than the regression task and so the benefit of weighting in favor of the factorization task has a greater return than the performance drop in the regression task. This also seems like a good place to perform tuning to find the best "balance" between these parameters.