‘Uncertainty’-based Continual Learning with ‘Adaptive’ Regularization

Ahn , Cha, Lee et al

Implementation, Reproducibility and Experiments
Mohamed Abdelsalam , Amir Raza

What is Continual Learning?

Continual learning: Data arrives from multiple tasks sequentially . Learning algorithm should adapt to new tasks ,while not forgetting what it learnt in the past.

Today: Uncertainty based Continual Learning (UCL)

-Determines important nodes. Doesn’t need to expand model for every task.

Uses: Bayesian online learning framework -ie **Posterior over the weights**

How to do Continual Learning?

Replaying past samples, Regularization based methods, Parameter isolation

- **Regularization-based** methods ‘**identify important weights’** . **Penalize large updates** on those weights, while learning a new task.
- Variational inference- learn approximation of posterior distribution on models.
- Obtain posterior $p(W|\alpha, D)$ after observing data D. Exact posterior intractable, variational inference approximates this posterior with a more tractable distribution $q(W|\theta)$.

Interpreting KL-divergence and motivation of UCL

Original Loss term (Free energy):.

$$\mathcal{F}(D,\boldsymbol{\theta}) = \mathbb{E}_{q(\boldsymbol{\mathcal{W}}|\boldsymbol{\theta})}[-\log p(D|\boldsymbol{\mathcal{W}})] + D_{KL}(q(\boldsymbol{\mathcal{W}}|\boldsymbol{\theta})||p(\boldsymbol{\mathcal{W}}|\boldsymbol{\alpha}))$$

Applied to Continual Learning, q(W| theta t-1) ~ P(W| theta):

$$\mathcal{F}(D_t,\boldsymbol{\theta}_t) = \mathbb{E}_{q(\boldsymbol{\mathcal{W}}|\boldsymbol{\theta}_t)}[-\log p(D_t|\boldsymbol{\mathcal{W}})] + D_{KL}(q(\boldsymbol{\mathcal{W}}|\boldsymbol{\theta}_t)||q(\boldsymbol{\mathcal{W}}|\boldsymbol{\theta}_{t-1}))$$

Use Gaussian Mean Field assumption : Loss reinterpreted in means and variance:

$$\frac{1}{2}\sum_{l=1}^{L}\Big[\underbrace{\Big\|\frac{\boldsymbol{\mu}_t^{(l)}-\boldsymbol{\mu}_{t-1}^{(l)}}{\boldsymbol{\sigma}_{t-1}^{(l)}}\Big\|_2^2}_{(a)} + \underbrace{\mathbf{1}^\top\Big\{\Big(\frac{\boldsymbol{\sigma}_t^{(l)}}{\boldsymbol{\sigma}_{t-1}^{(l)}}\Big)^2 - \log\Big(\frac{\boldsymbol{\sigma}_t^{(l)}}{\boldsymbol{\sigma}_{t-1}^{(l)}}\Big)^2\Big\}}_{(b)}\Big],$$

Finding important nodes from importance of weights

Memory is not unlimited !

Check if ‘**any node**’ in L or (L-1) is important

-Penalize update for strong connection

$$\frac{1}{2}\left(\sum_{l=1}^{L}\left\|\boldsymbol{\Lambda}^{(l)}\odot(\boldsymbol{\mu}_t^{(l)}-\boldsymbol{\mu}_{t-1}^{(l)})\right\|_2^2\right),\quad \text{where}\ \boldsymbol{\Lambda}_{ij}^{(l)}\triangleq\max\left\{\frac{\sigma_{\text{init}}^{(l)}}{\sigma_{t-1,i}^{(l)}},\frac{\sigma_{\text{init}}^{(l-1)}}{\sigma_{t-1,j}^{(l-1)}}\right\},$$

Final Loss:

$$-\log p(D_t \mid W)+\sum_{l=1}^{L}\Bigg[\frac{1}{2}||\Lambda^l\circ({\mu_t}^l-{\mu_{t-1}}^l)||_2^2+ ({\sigma_{init}}^l)^2||\left(\frac{{\mu_{t-1}}^l}{{\sigma_{t-1}}^l}\right)^2\circ({\mu_t}^l-{\mu_{t-1}}^l)||_1+ \frac{\beta}{2}.1^T\left[\left(\frac{{\sigma_t}^l}{{\sigma_{t-1}}^l}\right)^2-\log\left(\frac{{\sigma_t}^l}{{\sigma_{t-1}}^l}\right)^2+({\sigma_t}^l)^2-\log({\sigma_t}^l)^2\right]\Bigg]$$

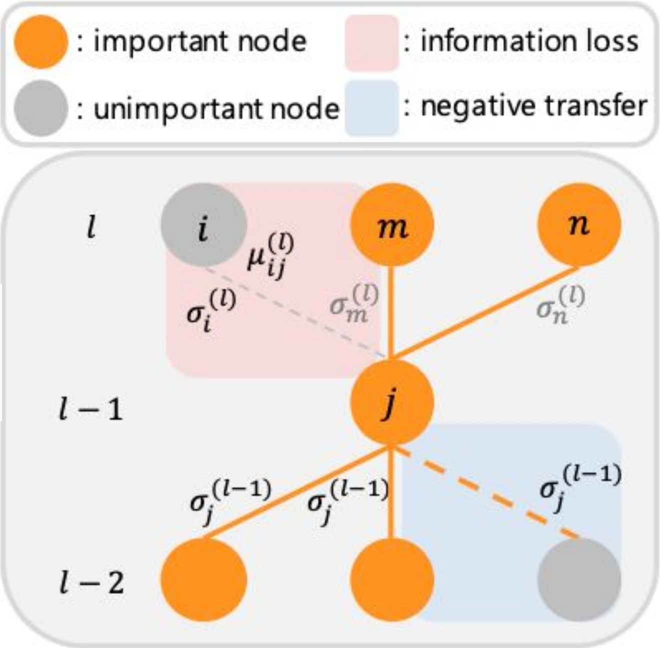
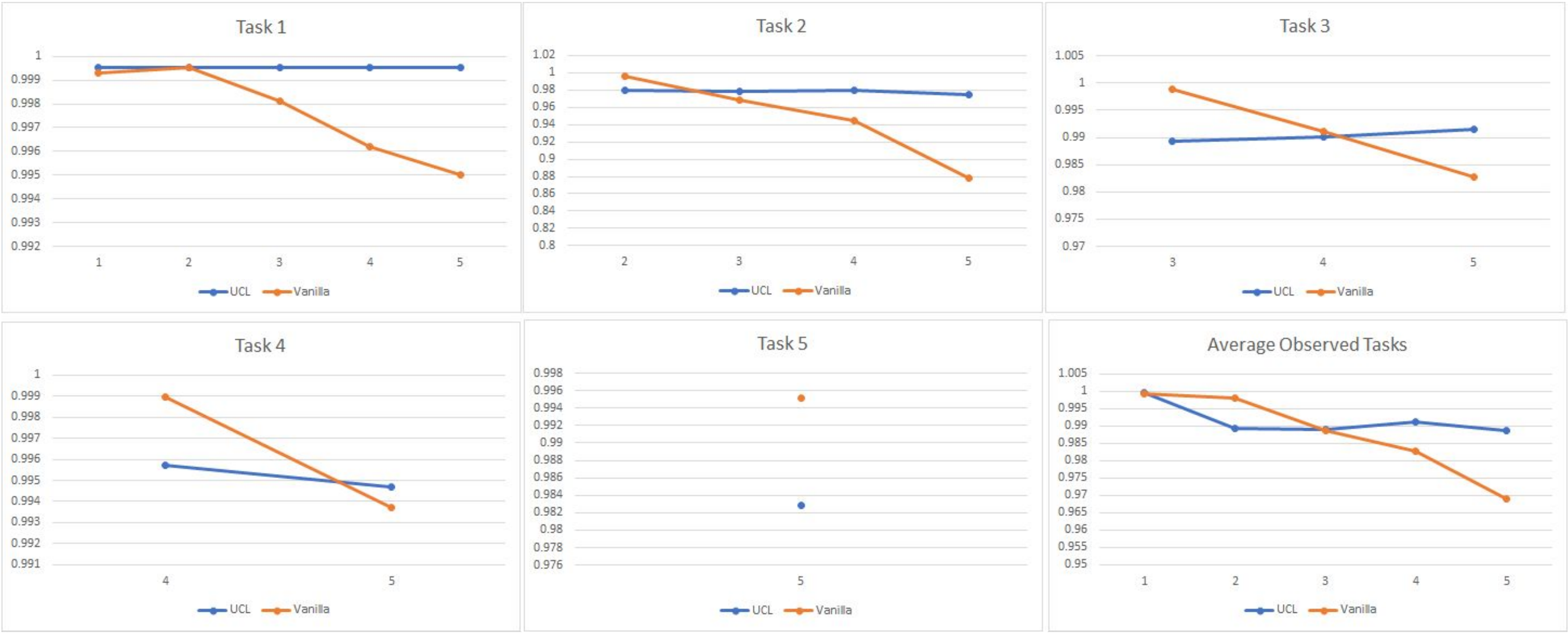


Figure 1: Information loss and negative transfer of an important node.

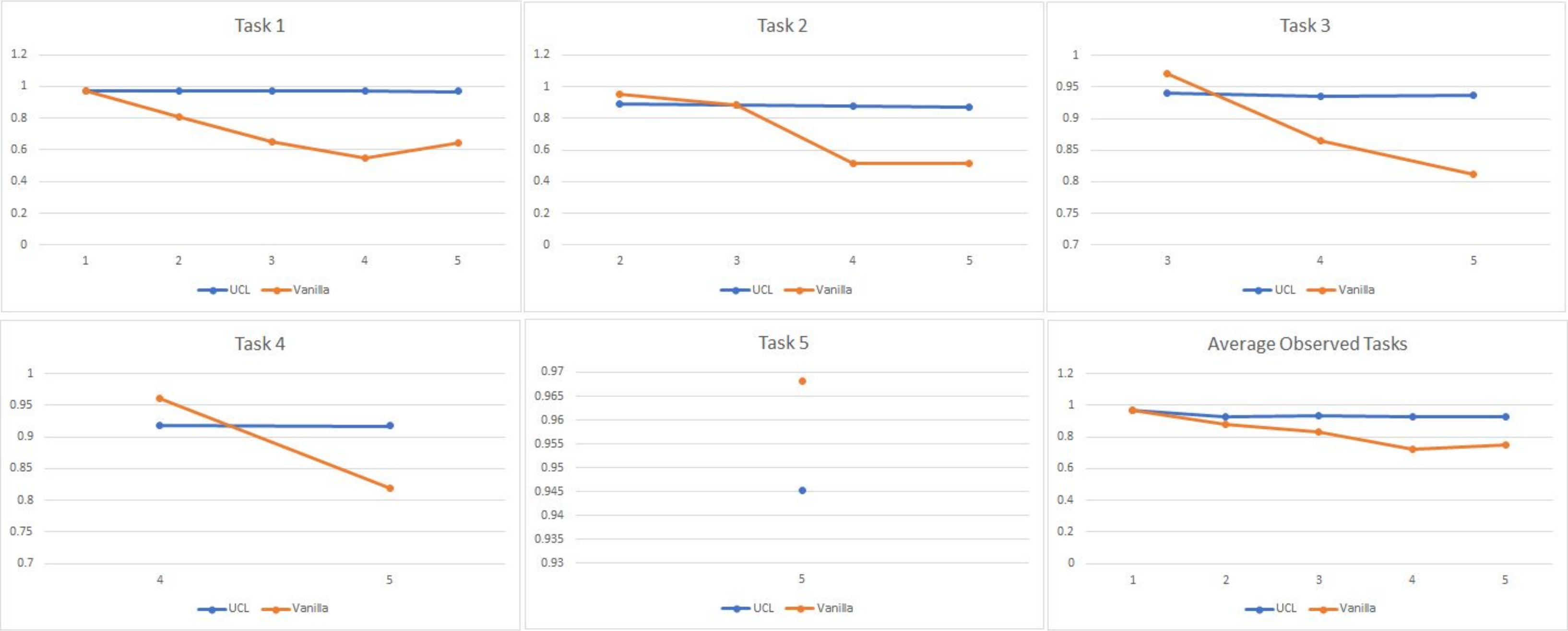
Mnist (UCL vs Vanilla)



Test accuracy on observed data sets (3 runs each)

Bottom Line: Average accuracy UCL 98.9%, Vanilla 96.9%
Reported Paper Results: VCL 98.7%. UCL 99.6%

NotMnist (UCL vs Vanilla)



Test accuracy on observed data sets (3 runs each)

Bottom Line: Average accuracy UCL 92.9%, Vanilla 75%
Reported Paper Results: EWC 84%, VCL 90.1%. UCL 95.7%

Observations and Conclusion

Regularizer is everything ! Don’t divide regularization term with minibatch

Bias is not Sampled to simplify the problem

Sigma_init appears like an arbitrary addition

How does model selection work?

Hyperparameters not mentioned in the paper (scheduler, regularization weight)

Final Verdict: It needs fine-tuning, but still beats the competitors!