Aalto University

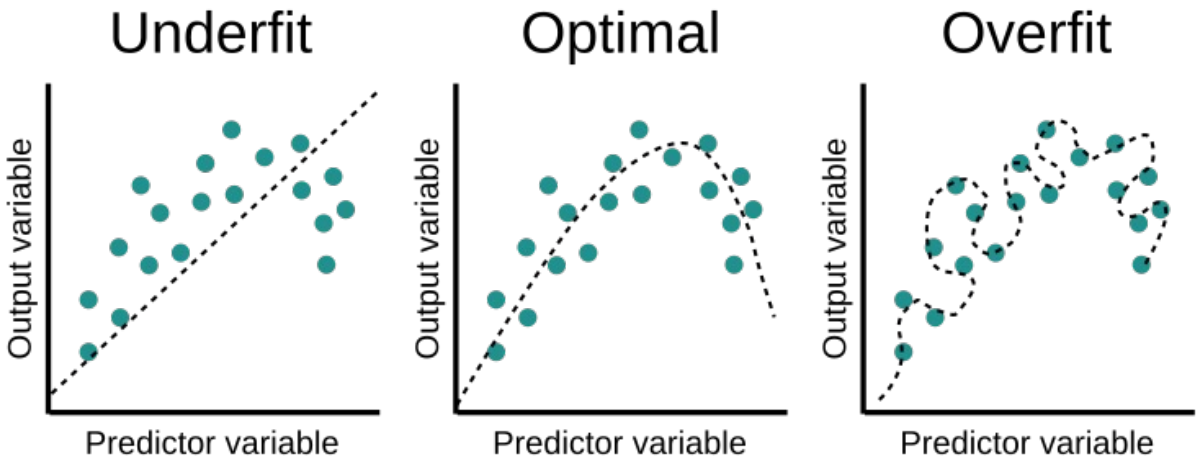
CS-E4890: Deep Learning

Regularization
Q&A session

Daolang Huang
17.03.2023

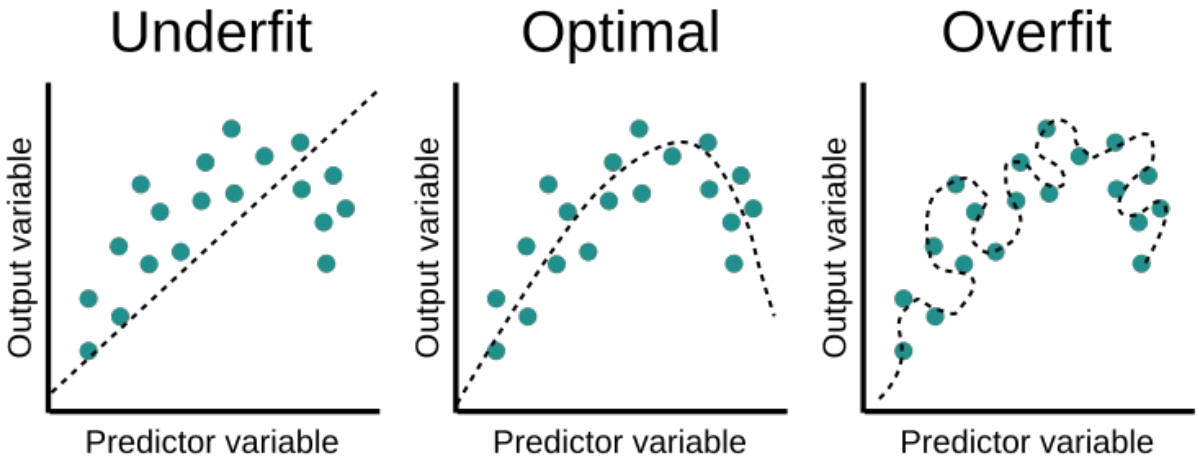
Overfitting

- Model fits the training data so closely that it fails to generalize to unseen data, resulting in good performance on the training set, but very poor performance on new data



Why it happens?

- Insufficient data samples
- The model is overtrained on a same dataset
- The model is too complex, so it learns the noise within the training data

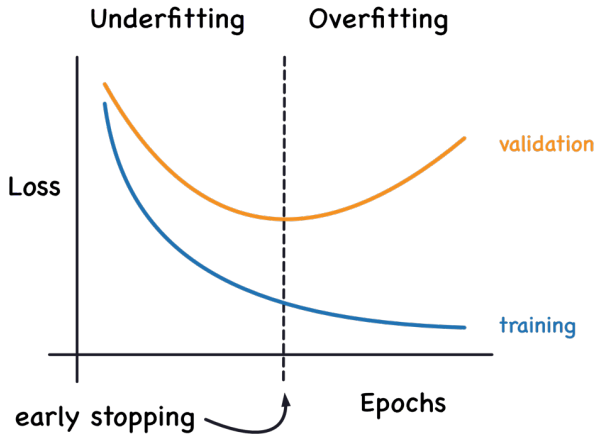


Regularization

- Early stopping
- Reduce model capacity
- Weight decay
- Noise injection
- Dropout

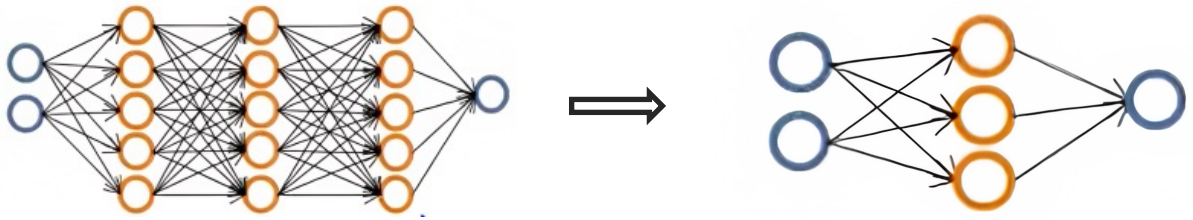
Early Stopping

- We need to create a validation dataset that the model has not been trained on
- Monitoring the performance of a model as it is trained, and stopping the training process when the model begins to show signs of overfitting



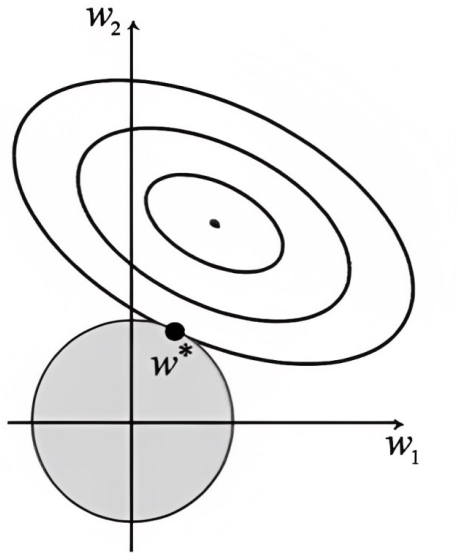
Reduce Model Capacity

- Limit the complex of model
- A less complex model has fewer parameters to learn, so it is less likely to fit the training data too closely
- Disadvantage: Sacrificing model prediction accuracy



Weight decay

- Penalize large weights in the model
- Smaller weights make the model less sensitive to small variations in the input data, resulting in a smoother function

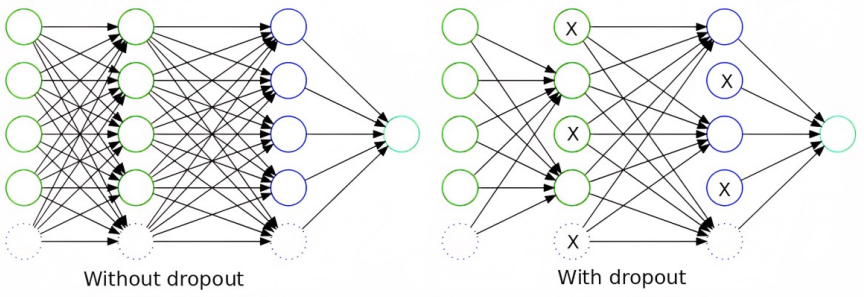


Noise Injection

- Introducing randomness and variability in the training data
- Can be seen as a data augmentation trick
- Reduce the chances of the model memorizing the data

Dropout

- Randomly dropping out neurons from the network during training
- Prevents the model from relying too heavily on any one neuron
- Use all neurons to do the prediction in the test stage, which can be seen as an ensemble method



Assignment 03_reg

- Exercise 1. Regularization techniques (0 point)
  - Implement all regularization methods mentioned previously
    - Early stopping
    - Weight decay
    - Noise injection
    - Dropout
    - Reduce model size
  - Solve the overfitting problem on a toy regression case

Assignment 03_reg

- Exercise 2. Recommender system (6 points)
  - Predict the user's rating of the movie given the `user_id` and the `item_id`
    - Input: embeddings of `user_id` + embeddings of `item_id`
    - Output: the predicted rating $\hat{r}$
  - Overfitting is a severe problem for this dataset
  - Target: Achieve low MSE loss on the test set with different regularization techniques

Assignment 03_reg

- `nn.embedding`
  - A lookup table that stores embeddings for a set of discrete input symbols
  - Similar users/items will have close representations $\boldsymbol{S}$ after training
    - If $\boldsymbol{S}_{user1}$ and $\boldsymbol{S}_{user2}$ are similar, in training data we have $r(\boldsymbol{S}_{user1}, \boldsymbol{S}_{movie1})$, it can help us predict $r(\boldsymbol{S}_{user2}, \boldsymbol{S}_{movie1})$
  - Equals to "`index to one-hot vector transformation`" + `nn.Linear(.,.,bias=False)`