QUESTION #1

- LASSO → $J(\beta) = \|y - X\beta\|^2 + \lambda\|\beta\|_1$
- $J_{TRAIN} << 1$
- $J_{val} >> 1$

a) **OVERFITTING**

b) 1) INCREASE NUMBER OF TRAINING SAMPLES
2) INCREASE REGULARIZATION PARAMETER λ

c) if Ridge is to replace $+ \lambda\|\beta\|_2^2$

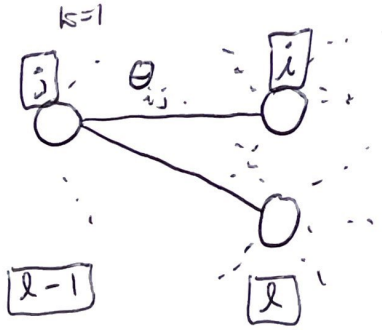
→ this could decrease the testing error because ridge constrains the size of the coefficients, shrinks some coefficients to zero but never completely/exactly to zero.

→ although it truly depends on the data set, one could create another validation set to compare these two approaches

QUESTION # 2

- L - layer network

$$J(\theta) = \sum_{k=1}^{n} \| y^{(k)} - a^{(k)(L)} \|^2 + \frac{\lambda}{2} \sum_{l} \sum_{i} \sum_{j} \theta_{ij}^{l\,2}$$



a) 2 solutions to improve the performance if the model is overfitting
- similar to the question above

1) we can increase the regularization parameter thus results in smaller weights which help with overfitting

2) we could decrease the complexity of the network / model. say decrease # of layers to L-3.

3) increase number of training samples

b) 2 solutions to improve the performance if the model is underfitting

1) on the other hand, if it underfitting we could increase the complexity by adding more layers or neurons. say L+3

2) Decrease the regularization parameter λ

QUESTION #3

- dropout method of regulation is sometimes preferred

a) **def:** Dropout involves randomly dropping out (setting to zero) some neurons in the network during training.

advantages:

- this results in remaining neurons having to work together to produce the output rather than a few specific neurons
- improved generalization
- prevent overfitting
- Easy to implement
- computationally efficient

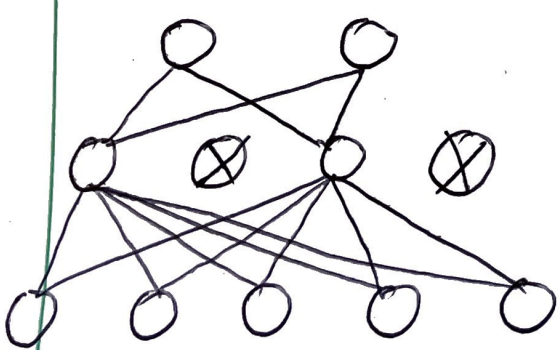
disadvantages

- increased training time
- sensitivity to dropout rate hyperparam
  - ↳ rate too high → underfit
  - ↳ rate too low → overfit
- reduced capacity can limit its ability to learn complex patterns

IT'S CONTROLLED VIA THE DROPOUT RATE HYPERPARAMETER

b)

w/ DROPOUT



w/o DROPOUT