Regularization Effect of Dropout

Xunyi Zhao
Supervisor: David Tax

Delft University of Technology

Regularization

Overfitting: when the model is too flexible, its training performance can be much better than the test performance

Regularization

Overfitting: when the model is too flexible, its training performance can be much better than the test performance

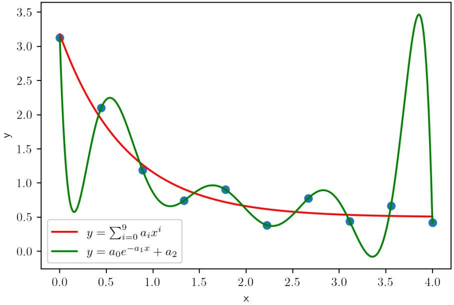


Figure: Fitting the same dataset with different functions

Dropout

Deep neural networks are flexible, need regularizers to prevent overfitting.

Dropout

Deep neural networks are flexible, need regularizers to prevent overfitting.

Dropout: randomly replace the outputs of some neurons as 0's during training.

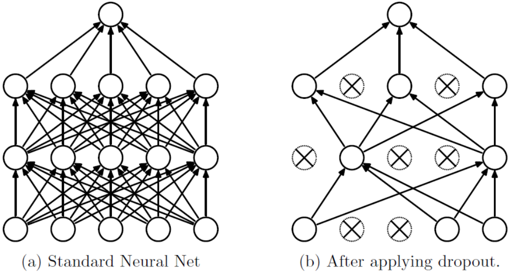


Figure: Srivastava N, Hinton G, Krizhevsky A, et al. *Dropout: a simple way to prevent neural networks from overfitting.* 2014

Dropout

How does the dataset size affect dropout's performance?

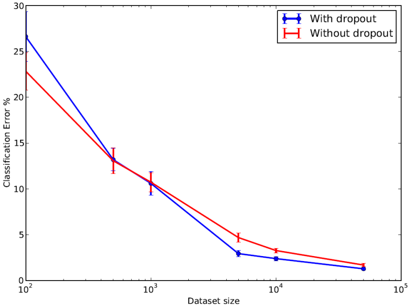


Figure: Srivastava N, Hinton G, Krizhevsky A, et al. *Dropout: a simple way to prevent neural networks from overfitting.* 2014.

Behaviors

Binary classification task, each class from a 10-d Gaussian distribution.
Generalization gap = Training accuracy - Test accuracy.

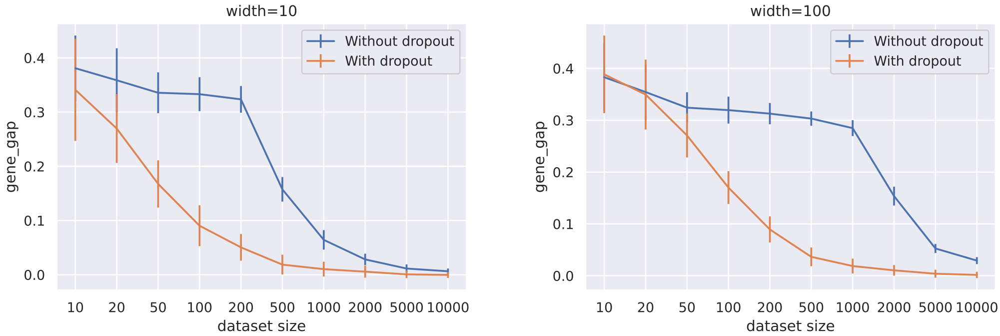


Figure: Left: (10-10-10-2) networks. Right: (10-100-100-2) networks

Behaviors

- Dropout doesn't work when the training set is too small or too large.
- Large networks need more training samples to make dropout work.

Behaviors

- **Dropout doesn't work when the training set is too small or too large.**
- **Large networks need more training samples to make dropout work.**

Complexity

Number of parameters, norm of weights, etc.

Complexity

Number of parameters, norm of weights, etc.

- **Model class**
  - e.g. Rademacher complexity
- **Specific model**
  - e.g. Norm of weights

Complexity

Number of parameters, norm of weights, etc.

- **Model class**
  - e.g. Rademacher complexity
- **Specific model**
  - e.g. Norm of weights

Effect of dataset sizes $\rightarrow$ complexity of specific models

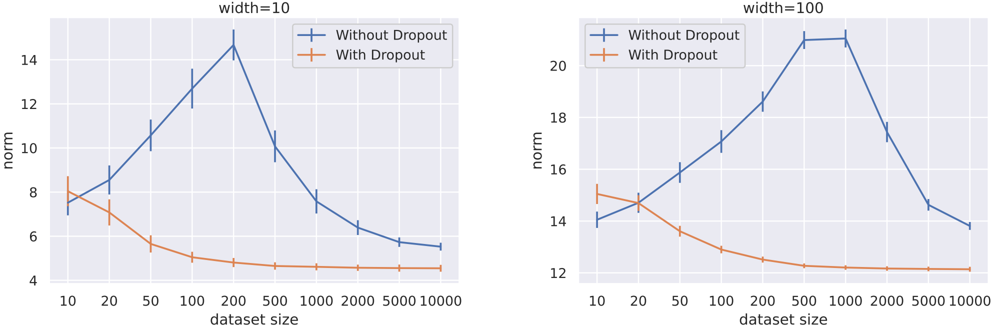
Complexity

Complexity measures for specific models:

- **Norm**: Frobenius norm of weights.
- **Sharpness**: Second-order derivative of loss with respect to weights.
- **Sensitivity**: Derivative of prediction with respect to the input data.

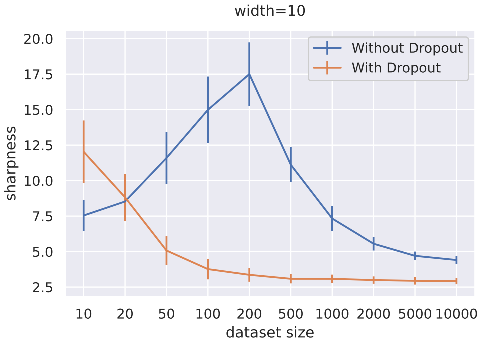
Complexity

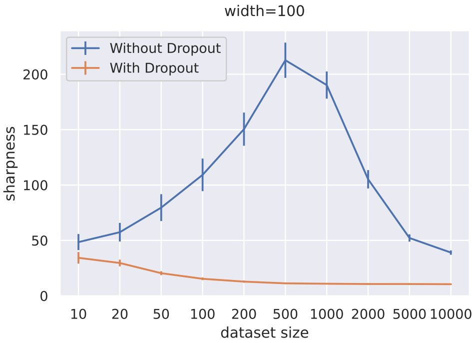
Norm: $\sum_l \left\|\theta^{(l)}\right\|_{\mathrm{F}}$, where $\|\cdot\|_{\mathrm{F}}$ is the Frobenius norm.



Complexity

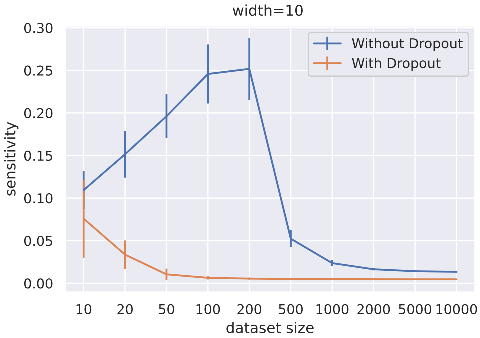
Sharpness: $\sum_l \sqrt{\|\theta^{(l)}\|_F^2 \, H^{(l)}}$, where $H^{(l)} := \sum_{i,j} \frac{\partial^2 \mathcal{L}(f_\Theta(X), Y)}{\partial \theta_{i,j}^{(l)} \partial \theta_{i,j}^{(l)}}$ [2]

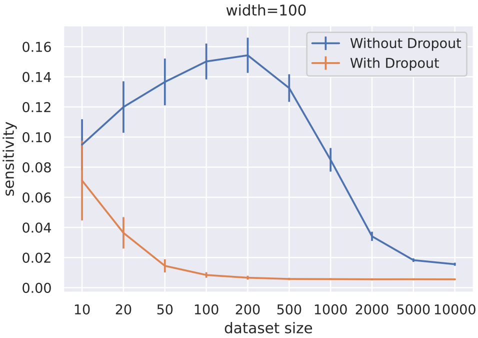




Complexity

Sensitivity: $E_X\left[\|\mathbf{J}(X)\|_{\mathrm{F}}\right]$, where $\mathbf{J}(X) = \partial f_{\Theta}(X)/\partial X^{\mathsf{T}}$[1]





Complexity

- Complexity is low when the dataset size is very small **and** very large.
- Large networks find maximum with larger datasets.
- **Dropout works when the model's complexity is high.**

Complexity

- Complexity is low when the dataset size is very small **and** very large.
- Large networks find maximum with larger datasets.
- **Dropout works when the model's complexity is high.**

Classification Boundary

Predict score over the input space:

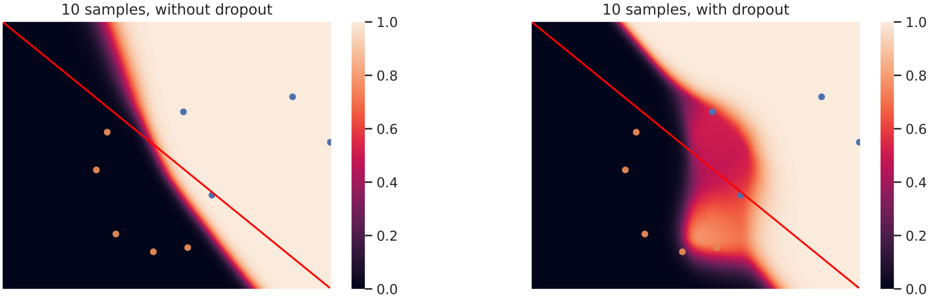


Figure: Prediction probability surface of the networks trained on the 2-d Gaussian dataset. Each axis represents one input feature range from [-3,3].

Classification Boundary

Predict score over the input space:

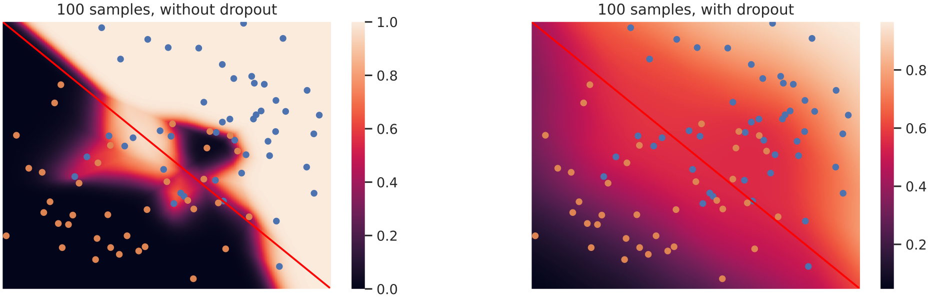


Figure: Prediction probability surface of the networks trained on the 2-d Gaussian dataset. Each axis represents one input feature range from [-3,3].

Classification Boundary

Predict score over the input space:

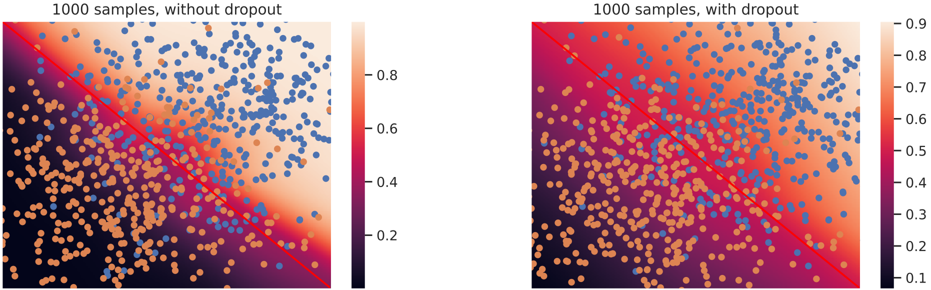


Figure: Prediction probability surface of the networks trained on the 2-d Gaussian dataset. Each axis represents one input feature range from [-3,3].

Classification Boundary

- When the dataset is small, overfitting the samples does not require a complex boundary.
- Only when the dataset size is reasonably large, the model can be very complex to fit all the samples.
- When there are too many samples, a simple boundary would again be the best choice.

Classification Boundary

- When the dataset is small, overfitting the samples does not require a complex boundary.
- Only when the dataset size is reasonably large, the model can be very complex to fit all the samples.
- When there are too many samples, a simple boundary would again be the best choice.

Neuron Loss

Assumption: neurons have to work together to create a complicated boundary.

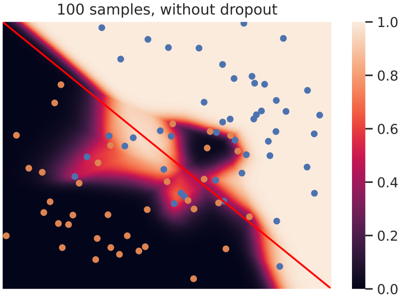


Figure: Prediction probability surface of the networks trained on the 2-d Gaussian dataset. Each axis represents one input feature range from [-3,3].

Neuron Loss

Train without dropout, test with dropout.

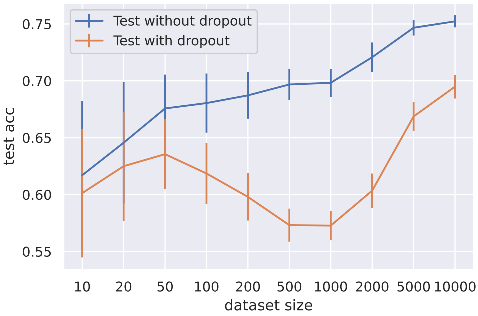


Figure: Test accuracy vs. dataset sizes. Dropout is only applied in the test phase.

Neuron Loss

- When neurons work together to create a complex boundary, it is vulnerable to neuron loss.
- If a model is not sensitive to neuron loss, applying dropout would not make a difference.

Neuron Loss

- When neurons work together to create a complex boundary, it is vulnerable to neuron loss.
- If a model is not sensitive to neuron loss, applying dropout would not make a difference.

Conclusion

- **Dropout doesn't work when the training set is too small or too large.**
- **Large networks need more training samples to make dropout work.**

Conclusion

- **Dropout doesn't work when the training set is too small or too large.**
- **Large networks need more training samples to make dropout work.**
- The model's complexity is high when the dataset is reasonably large.
- Dropout works when the model's complexity is high.

Conclusion

- **Dropout doesn't work when the training set is too small or too large.**
- **Large networks need more training samples to make dropout work.**

- The model's complexity is high when the dataset is reasonably large.
- Dropout works when the model's complexity is high.

References I

Roman Novak et al. 'Sensitivity and generalization in neural networks: an empirical study'. In: *arXiv preprint arXiv:1802.08760* (2018).

Yusuke Tsuzuku, Issei Sato and Masashi Sugiyama. 'Normalized flat minima: Exploring scale invariant definition of flat minima for neural networks using pac-bayesian analysis'. In: *International Conference on Machine Learning*. PMLR. 2020, pp. 9636–9647.