SLIDE 1-INTRODUCTION:

Introduction to overfitting in supervised training of classification and regression functions: In supervised training, the goal is to learn a function that maps input data to corresponding output labels or values. This function is trained using a dataset that consists of input-output pairs. The objective is to minimize the discrepancy between the predicted outputs and the true outputs in the training data.

However, a common challenge in supervised training is overfitting. Overfitting occurs when the learned function becomes too complex and fits the training data too closely, to the point that it fails to generalize well to new, unseen data. In other words, the model becomes excessively tailored to the training dataset and loses its ability to make accurate predictions on new examples.

Discrepancy between training error and test error: To evaluate the performance of a trained model, we typically measure its error on both the training data and a separate set of data called the test or validation data. The training error is calculated by comparing the model's predictions on the training data to the true labels or values. The test error, on the other hand, measures the model's performance on the unseen test data.

In ideal scenarios, we expect the training error and test error to be similar, indicating that the model has learned the underlying patterns and can generalize well to new data. However, in cases of overfitting, the training error becomes significantly lower than the test error. This discrepancy arises because the model has become too specific to the training data, capturing even the noise or random fluctuations in the data. As a result, when faced with new data, the overfitted model fails to generalize and exhibits higher error rates.

The presence of a significant gap between the training and test error is a clear indication of overfitting. It suggests that the model has memorized the training examples rather than learning the underlying patterns that enable accurate predictions. To combat overfitting, various techniques, including regularization, cross-validation, and early stopping, are employed to find the right balance between model complexity and generalization ability.

Slide -2 : Regularization and Bayesian Framework

Regularization term as a countermeasure against overfitting: When training a supervised model, overfitting can occur when the model becomes too complex and captures noise or irrelevant patterns in the training data. To mitigate overfitting, a regularization term is often added to the objective function during training. The regularization term helps control the complexity of the model by imposing a penalty on certain model parameters.

By introducing the regularization term, the optimal parameter values for the objective function become less dependent on the likelihood term alone. The regularization term serves as a constraint that encourages the model to find a balance between fitting the training data well and avoiding excessive complexity. This regularization helps prevent the model from overemphasizing noise or idiosyncrasies in the training data, leading to improved generalization performance on unseen data.

Regularization term as the logarithm of prior distribution in Bayesian framework: In the Bayesian framework, a prior distribution is defined over the model parameters, representing our beliefs or preferences about the parameter values before observing any data. By incorporating a prior distribution, we can express our prior knowledge or assumptions about the parameters and guide the model towards plausible solutions.

In the context of regularization, the regularization term can be seen as the logarithm of the prior distribution. The regularization term influences the model by defining the preference or bias towards certain parameter values. By incorporating the prior distribution, we shape the model's behavior, favoring parameter settings that align with our prior beliefs. This helps regularize the model and prevents it from overfitting by constraining the parameter space.

Importance of choosing a good model distribution: The choice of a model distribution is crucial in solving a specific problem effectively. While there is no universally good model distribution that applies to all scenarios, the selection of a suitable model distribution should be based on the characteristics of the problem at hand.

Different problems may exhibit different underlying structures or patterns in the data. For example, natural images and time series often exhibit smoothness in their spatial or temporal dimensions. By understanding the problem domain, we can make informed decisions about the desired properties of the model distribution.

Choosing an appropriate model distribution allows us to capture the inherent characteristics of the data, making the model more expressive and capable of generalizing well. It enables the model to capture the relevant features and relationships in the data, leading to improved performance in prediction or classification tasks.

Preference for smooth outputs in good models: In many real-world applications, smoothness is considered a desirable property in the outputs of good models. Smoothness implies that small changes in the input data lead to gradual changes in the model's predictions. This property aligns with our intuition about how the data should behave in the problem domain.

For example, in image processing or time series analysis, we expect nearby data points to have similar values or labels. Smoothness in the outputs ensures that the model's predictions exhibit a coherent and continuous behavior, reflecting the underlying structure of the data.

By incorporating a preference for smoothness in the model distribution, we encourage the model to produce outputs that adhere to this characteristic. This regularization towards smoothness helps prevent the model from generating overly noisy or erratic predictions, leading to more reliable and interpretable results.

Overall, by understanding the problem domain and incorporating the preference for smoothness in the model distribution, we can improve the model's performance and enhance its ability to generalize well to new, unseen data.

Slide -3: Local Distributional Smoothness (LDS) Regularization

Description of LDS as a novel regularization term: Local Distributional Smoothness (LDS) is introduced as a novel regularization term that aims to improve the smoothness of the model distribution with respect to the input data. Traditional regularization techniques focus on controlling the complexity of the model, but LDS specifically targets the local smoothness properties of the model distribution.

Rewarding the smoothness of the model distribution with respect to input: The objective of LDS is to encourage the model to produce outputs that exhibit smoothness characteristics concerning the input data. Smoothness implies that small changes in the input should result in gradual changes in the model's predictions. By rewarding the smoothness of the model distribution, the goal is to improve the model's ability to capture the underlying structure and relationships present in the data.

Parametrization invariance of the regularization term: One key advantage of LDS is its parametrization invariance property. Regardless of the chosen parameterization of the model, the regularization term remains consistent. This property ensures that the optimal model distribution obtained through training with LDS regularization is unique and independent of the specific parameterization used.

In other words, if we find the optimal parameters θ^* that maximize the objective function with LDS regularization, and then apply a diffeomorphism (a smooth and invertible transformation) to these parameters, the resulting model distribution will still be optimal in terms of the objective function. This invariance ensures the stability and uniqueness of the trained model, regardless of the parameterization chosen.

This property is advantageous compared to other regularization methods like Lq regularization, which may lead to different optimal solutions depending on the parameterization. Additionally, for complex models like deep neural networks, assessing the effect of traditional regularization terms on the topology of the model distribution can be challenging, while LDS provides a more straightforward approach to promoting local smoothness.

Slide 4: VAT (Virtual adversial training)

How VAT Works

VAT works by adding a penalty to the model's loss function that encourages the model to be more robust to small changes in the input data. VAT does this by generating adversarial examples. Adversarial examples are small, carefully crafted perturbations of input data that can cause a model to make incorrect predictions.

VAT generates adversarial examples by adding random noise to the input data and then using the model to predict the labels of the noisy data. The model's predictions for the noisy data are then used to update the model's parameters. This process is repeated for a number of iterations, until the model becomes more robust to adversarial examples.

Advantages of VAT

VAT has a number of advantages over other regularization methods. It is:

- Effective in preventing overfitting
- Can be used with both supervised and semi-supervised learning
- Has a small number of hyperparameters, which makes it easy to tune
- Parametrization invariant, which means that it works with any type of machine learning model
- Computationally efficient

Disadvantages of VAT

VAT also has a few disadvantages. It can be:

- Slow to converge
- Can reduce the accuracy of the model on the training data

Slide 5: Relationship with Adversarial Training

Comparison with adversarial training by Goodfellow et al.: Virtual Adversarial Training (VAT), based on the concept of Local Distributional Smoothness (LDS), is compared to adversarial training proposed by Goodfellow et al. While both methods involve perturbing the input data, there are key differences in their approaches.

Adversarial training by Goodfellow et al. focuses on identifying the direction of input perturbations that maximally affect the model's label assignment for a given input. The model is then penalized for its sensitivity to these perturbations. In contrast, VAT does not rely on label information and instead measures the model's robustness against both local and "virtual" adversarial perturbations.

Applicability of VAT to both supervised and semi-supervised learning: Virtual Adversarial Training (VAT) is applicable to both supervised and semi-supervised learning settings. In supervised learning, VAT can be used to train models using labeled data, where the objective function includes the log likelihood of the dataset augmented with the sum of LDS computed for each input data point.

Additionally, VAT can be extended to semi-supervised learning, where labeled and unlabeled data are available. The advantage of VAT in semi-supervised learning is that it does not require label information for calculating the regularization term. This allows the method to leverage the unlabeled data effectively and improve the model's performance in the semi-supervised setting.

Advantage of not requiring label information: One significant advantage of VAT is that it does not depend on label information when calculating the regularization term. Traditional adversarial training relies on the correct labels to determine the adversarial directions, which limits its applicability to supervised learning scenarios.

In VAT, the virtual adversarial perturbations are determined without the need for labels. This property makes VAT suitable for semi-supervised learning, where unlabeled data can be utilized to improve the model's robustness and generalization. By not requiring label information, VAT offers a more flexible and versatile regularization method that can be applied in a broader range of learning scenarios.

Slide 6: Efficient Gradient Approximation for LDS

Second-order Taylor expansion of LDS: To efficiently approximate the gradient of the Local Distributional Smoothness (LDS), a second-order Taylor expansion is employed. By expanding the LDS function around a specific point, higher-order derivatives can be approximated, which leads to a more accurate estimation of the gradient.

The second-order Taylor expansion provides an approximation of the LDS function that takes into account not only the first-order derivative (gradient) but also the second-order derivatives (Hessian matrix) of the model distribution with respect to the input data.

Application of power method for efficient gradient approximation: To approximate the gradient of the LDS efficiently, the power method is utilized. The power method is an iterative algorithm that estimates the dominant eigenvector of a matrix, which in this case is the Hessian matrix of the model distribution.

By applying the power method, the dominant eigenvector, which corresponds to the gradient direction, can be efficiently computed without directly computing the Hessian matrix itself. This approach reduces the computational complexity and allows for a more efficient approximation of the gradient.

Low computational cost with only three pairs of forward and back propagations: One of the advantages of the Virtual Adversarial Training (VAT) method is its low computational cost. Specifically, to compute the approximate gradient of the LDS, only three pairs of forward and backward propagations are required.

This means that for each input data point, the model needs to be evaluated three times: one forward pass to compute the initial model distribution, and two additional passes to calculate the forward and backward perturbations needed for the power method. The gradients estimated from these three passes are then used to update the model parameters.

This computational efficiency makes VAT particularly suitable for complex models such as deep neural networks, as it provides a computationally affordable regularization method that can be applied during training without significantly increasing the training time.

Slide 7:

two synthetic datasets: "Moons" and "Circles". These datasets were designed to test the performance of different regularization methods.

- The "Moons" dataset consists of data points arranged in the shape of two intersecting half circles.
- The "Circles" dataset consists of data points arranged in the shape of two concentric circles.

For each dataset, we had a total of 16 training samples and 1000 test samples. The limited number of training samples allowed us to examine the impact of regularization methods on small data scenarios.

The primary goal of our experiments was to evaluate the effectiveness of various regularization methods in improving the generalization performance of neural networks. By comparing the results on these synthetic datasets, we aimed to gain insights into the strengths and weaknesses of different regularization techniques.

In our experiments, we employed several regularization methods to prevent overfitting in neural networks. These methods included:

1. L2 Regularization: Also known as weight decay, L2 regularization adds a penalty term to the loss function that discourages large weights. This helps to prevent over-reliance on individual features and encourages the model to generalize better.
2. Dropout: Dropout randomly deactivates a fraction of neurons during training, forcing the network to learn redundant representations. This helps to reduce co-adaptation among neurons and improves the robustness of the network to noise.
3. Adversarial Training: Adversarial training introduces small perturbations to the input data to make the model more robust against adversarial attacks. It involves iteratively finding perturbations that maximize the model's loss, forcing the model to learn more generalized decision boundaries.
4. Random Perturbation Training: Random perturbation training involves adding random noise to the input data during training. This serves as a form of regularization by introducing variability and reducing overfitting.
5. Virtual Adversarial Training (VAT): VAT is a novel regularization method that rewards the smoothness of the model distribution with respect to input

perturbations. It uses the concept of local distributional smoothness to improve the generalization performance of neural networks.

On the "Moons" dataset:

- L2 Regularization: Test Error = 0.20
- Dropout: Test Error = 0.17
- Adversarial Training: Test Error = 0.11
- Random Perturbation Training: Test Error = 0.18
- VAT: Test Error = 0.09

On the "Circles" dataset:

- L2 Regularization: Test Error = 0.14
- Dropout: Test Error = 0.12
- Adversarial Training: Test Error = 0.08
- Random Perturbation Training: Test Error = 0.12
- VAT: Test Error = 0.07

It is evident from these results that both adversarial training and VAT achieved the lowest test errors on both datasets. Adversarial training and VAT consistently outperformed the other regularization methods in terms of generalization performance. This suggests that these methods effectively prevented overfitting and learned more robust decision boundaries.

The superior performance of adversarial training and VAT highlights their effectiveness in improving the generalization capability of neural networks. These methods introduce additional constraints and promote smoother model distributions, leading to better performance on unseen data. Their ability to achieve lower test errors on both the "Moons" and "Circles" datasets demonstrates their potential as powerful regularization techniques.

- Implications and findings: The results for the "Moons" dataset .i. This dataset is a challenging dataset for neural networks, because the data points are not linearly separable. The fact that VAT was able to achieve a low test error on this dataset suggests that VAT is effective at learning non-linear decision boundaries.
- The results for the "Circles" dataset. This dataset is a simpler dataset for neural networks, because the data points are linearly separable. The fact that VAT was able to achieve a lower test error than L2 regularization and dropout on this dataset suggests that VAT is

more effective at preventing overfitting than these other regularization methods.

- Overall, the results of the experiments that you have described suggest that VAT is a promising regularization method that can be used to improve the generalization performance of neural networks.

Slide 8: SUPERVISED LEARNING FOR THE CLASSIFICATION OF THE MNIST DATASET

This slide presents an overview of the experimental setup conducted on the MNIST dataset. It outlines the key aspects of the experiment, including dataset characteristics, data split, neural network architecture, activation function, and batch normalization.

The MNIST dataset is described, emphasizing that it consists of 28x28 pixel images of handwritten digits and their corresponding labels. The input dimension is calculated as 28 x 28 = 784, and the labels range from 0 to 9.

The dataset is split into training, validation, and test sets. Specifically, out of the original 60,000 training samples, 50,000 are used for training, and 10,000 are reserved for validation to tune the hyperparameters.

Two types of neural networks are trained: one with 2 layers and the other with 4 layers. The number of hidden units varies for each network configuration: (1200, 600) for the 2-layer network and (1200, 600, 300, 150) for the 4-layer network.

The ReLU activation function and batch normalization technique are employed across all neural networks to enhance their performance and training stability.

The regularization methods, including VAT, are applied to the training process using the hyperparameters that achieved the best performance on the validation set. The trained networks are then evaluated on the test set, with test errors recorded.

To ensure the reliability of the results, the experimental procedure is repeated 10 times with different weight initialization seeds. The average test error values are reported for each regularization method.

Table 1 summarizes the test error performance of VAT and other regularization methods. VAT demonstrates superior performance compared to contemporary methods, with the exception of Ladder network, which is a highly advanced generative model-based method.

Overall, this slide highlights the experimental setup conducted on the MNIST dataset and highlights VAT's strong performance in comparison to other regularization methods, indicating its potential as an effective regularization technique for improving neural network performance.

Slide 9: Experimental Results

The Virtual Adversarial Training (VAT) method has been applied to both supervised and semi-supervised learning tasks on the MNIST dataset, which is a popular benchmark dataset for image classification. In these tasks, VAT has shown promising results and has outperformed many contemporary methods, demonstrating its effectiveness in improving model performance.

However, it is worth noting that VAT did not surpass the performance of a state-of-the-art method developed by Rasmus et al. This state-of-the-art method likely utilizes a highly advanced generative model-based approach, which may have achieved superior results compared to VAT.

Furthermore, VAT has also been applied to other datasets such as SVHN and NORB in the context of semi-supervised learning. In these experiments, VAT has demonstrated superior performance compared to the current state-of-the-art semi-supervised methods applied to these datasets. This showcases the effectiveness and versatility of VAT across different datasets and learning scenarios.

Overall, the application of VAT to supervised and semi-supervised learning tasks on various datasets has shown promising results and has positioned VAT as a competitive regularization method that can enhance model performance and surpass existing methods in certain scenarios.

Slide 10: DISCUSSION AND RELATED WORKS

This slide focuses on comparing VAT with several existing regularization methods and shedding light on their similarities, differences, and performance.

The slide starts by mentioning that VAT was motivated by adversarial training and shares similarities with it. Both methods utilize the local input-output relationship to smooth the model distribution in the corresponding neighborhood, enhancing robustness and generalization.

In contrast, L2 regularization is highlighted as a method that does not use the local input-output relationship. It is explained that increasing the regularization constant in L2 regularization leads to global smoothing of the distribution, which may result in higher training and generalization errors.

The PEA method (Bachman et al., 2014) is introduced as an approach that aims to make the model distribution robust against random perturbations. However, it is noted that VAT outperforms both PEA and random perturbation training. The significance of the role of the Hessian matrix (H) in VAT is emphasized, as VAT projects perturbations in the principal direction of H, which aligns with the sensitivity of the distribution.

The Deep Contractive Network (Gu & Rigazio, 2015) is briefly mentioned as another method that smooths the model distribution. However, it is stated that this approach did not significantly decrease the test error.

The Ladder Network (Rasmus et al., 2015) is highlighted as the current state-of-the-art method for supervised and semi-supervised learning for the permutation invariant MNIST task. Its utilization of layer-wise denoising autoencoders for manifold learning is mentioned. VAT is contrasted with Ladder Network, stating that VAT focuses solely on the conditional distribution $p(y|x, \theta)$ without considering the generative process $p(x|\theta)$ or the full joint distribution $p(y, x|\theta)$. This highlights the complementary nature of VAT with methods that explicitly model the input distribution.

The slide concludes by acknowledging the potential for further improvement in VAT by incorporating notions of manifold learning into its framework. This indicates future research directions to enhance the performance and capabilities of VAT.

Overall, this slide provides a comprehensive comparison of VAT with existing regularization methods, highlighting its superior performance in certain cases and discussing opportunities for further development

Slide 11-CONCLUSION:

To summarize the experimental findings, our research demonstrates that VAT (Virtual Adversarial Training) is an effective regularization method for both supervised and semi-supervised learning. The experiments conducted on synthetic and real-world datasets, including MNIST, SVHN, and NORB, have provided compelling results.

Firstly, it is important to highlight the effectiveness of VAT. In comparison to other contemporary methods, VAT outperformed all of them on the MNIST dataset, except for the current state-of-the-art Ladder Network. This indicates that VAT is a powerful tool for improving the performance and generalization capabilities of neural networks.

Furthermore, VAT showcased superior performance on different datasets. Not only did it outperform other methods on MNIST, but it also surpassed the state-of-the-art semi-supervised learning method for SVHN and NORB datasets. This demonstrates that VAT's benefits extend beyond specific datasets and have broader applicability in various domains.

In addition to its effectiveness, VAT is also valued for its simplicity. With our approximation of the local distribution smoothing (LDS), VAT can be computed with reasonable computational cost. This makes it practical and feasible to implement VAT in real-world scenarios.

Another advantage of VAT is its computational efficiency. Compared to methods heavily reliant on generative models, VAT requires fewer computational resources. This allows for faster training and inference times, making it suitable for large-scale datasets and real-time applications.

Moreover, VAT stands out for its minimal hyperparameters. While some complex methods require fine-tuning of numerous hyperparameters, VAT demonstrates satisfactory results by optimizing only one hyperparameter ( ), with λ fixed at 1. This simplicity reduces the burden of hyperparameter tuning and enhances the ease of implementation.

In conclusion, the experimental findings affirm VAT's effectiveness as a regularization method, its superior performance across different datasets, its computational efficiency, and its minimal reliance on hyperparameters. These characteristics make VAT a promising approach for improving the generalization performance of neural networks in various real-world applications.