

# SEMI-SUPERVISED REGRESSION WITH GENERATIVE ADVERSARIAL NETWORKS FOR END TO END LEARNING IN AUTONOMOUS DRIVING

**Anonymous authors**

Paper under double-blind review

## ABSTRACT

This research concerns solving the semi-supervised learning problem with generative adversarial networks for regression. In contrast to classification, where only a limited number of distinct classes is given, the regression task is defined as predicting continuous labels for a given dataset. Semi-supervised learning is of vital importance for the applications where a small number of labeled samples is available, or labeling samples is difficult or expensive to collect. A case in point is autonomous driving in which obtaining sufficient labeled samples covering all driving conditions is costly. In this context, we can take advantage of semi-supervised learning techniques with groundbreaking generative models, such as generative adversarial networks. However, almost all proposed GAN-based semi-supervised techniques in the literature are focused on solving the classification problem. Hence, developing a GAN-based semi-supervised method for the regression task is still an open problem. To address this problem, we introduce Reg-GAN in two different architectures. In summary, our proposed method is able to predict continuous labels for a training dataset which has only a limited number of labeled samples. Moreover, the application of this technique for solving the end-to-end task in autonomous driving will be presented. We performed several experiments on a publicly available driving dataset to evaluate our proposed method, and the results are very promising. The results show that our approach generates images with high quality, gives smaller label prediction error and leads to a more stable training compared with the state-of-the-art Improved GAN technique (Salimans et al., 2016).

## 1 INTRODUCTION

*Autonomous driving* (AD) has gained attention from researchers and industry in the recent years. AD brings different fields of research such as computer vision, machine learning, and system engineering together. To make the driving task autonomous, the AD system should replace human being, which implies that the system should be able to recognize its peripheral environment and act accordingly. Machine learning can facilitate this task for the AD system. Machine learning is employed for different purposes in AD systems such as imitating driver behavior (Kuefler et al., 2017), vehicle detection (Huval et al., 2015), lane detection (Huval et al., 2015), and end to end learning (Bojarski et al., 2016). For example in (Bojarski et al., 2016), an end to end learning for self-driving cars was proposed where the convolutional neural network (CNN) is used to learn road features. In this regard, a comprehensive dataset of road images and corresponding steering angles is employed. A CNN is trained to map raw pixels from a single front facing camera directly to steering commands in a supervised manner. This CNN is able to learn meaningful road features from a very sparse training signal (the steering angle) without needing to decompose the front camera image into the road, lane marking detection, semantic abstraction, path planning, or control.

The importance of the training data is undeniable for machine learning tasks. For supervised learning algorithms, the training set should come with appropriate labels as well. Lack of enough labeled samples for supervised training leads to poor learning. On the other hand, collecting enough training samples and labeling them could be time consuming, difficult and costly. The cost of data collection is different depending on the application. As a case in point, training autonomous driving systems needs

a large number of training samples to guarantee that the system could learn all possible scenarios such as different driving states, road conditions, and weather conditions. Failing to learn a specific case by the AD system may result in safety problems. To tackle the issue of scarcity of the data, the following solutions have been proposed in the literature: first, using simulated samples and synthetic annotations rather than real datasets (Johnson-Roberson et al., 2016); second, using generative models for generating samples from a small unlabeled training set (Ghosh et al., 2016). However, the former suffers from producing unrealistic samples and poor performance in real applications, and the latter is not able to predict labels for the generated samples. The focus of this research is on the second approach, that is enriching the training dataset using generative models. To address the mentioned labeling issue of generative models, these models can be employed in a semi-supervised scheme, in which a small labeled portion of the training data can be leveraged for labeling the generated samples.

Semi-supervised learning algorithms are able to target label prediction problems when only a limited subset of training data is labeled (Kingma et al., 2014). In this regard, many semi-supervised learning techniques take advantage of deep generative models. Generative models aim at estimating the probability distribution of the training data and being able to generate samples which belong to the same data distribution manifold (Banijamali et al., 2017). Different methods for deep generative networks are proposed in the literature such as Deep Belief Networks (DBNs) (Hinton et al., 2006), Restricted Boltzman Machines (RBMs) (Salakhutdinov et al., 2007), Variational Auto-encoders (VAEs) (Kingma & Welling, 2013), and Generative Adversarial Networks (GANs) (Goodfellow et al., 2014); among them, GANs are the most recent and successful in generating realistic and good quality images (Arjovsky & Bottou, 2017).

This research concerns semi-supervised learning with generative adversarial networks for an end-to-end task in autonomous driving. Semi-supervised methods based on GANs have shown promising and competitive classification results as compared to traditional techniques (Salimans et al., 2016). In the literature, there are some semi-supervised techniques with GANs such as Improved-GAN (Salimans et al., 2016), Cat-GAN (Springenberg, 2015), SGAN (Odena, 2016), and Triple-GAN (Li et al., 2017); however, they all focus on solving the classification problem. On the other hand, the goal of the end-to-end task in AD is to predict the steering angle, which is a continuous variable, based on the given input image from the front camera.

Applying semi-supervised classification techniques to regression comes with the price of converting continuous labels of the dataset to a limited number of classes. This conversion will add the quantization error to our training, and also determining the number of classes for each application is non-trivial. Moreover, usually classification techniques require more number of outputs and so more number of network parameters comparing with regression techniques, which cause more computations and longer training time. Hence, using semi-supervised classification techniques for regression in the end-to-end task of AD is not easy and will increase the training cost. To the best of our knowledge, a semi-supervised regression technique with generative adversarial network yet has to be developed which is the main focus of this research. The main contributions of this paper are summarized as following:

1. To the best of our knowledge, we introduce the first semi-supervised algorithm with generative adversarial networks which can address the regression problem. We call our method Reg-GAN throughout the paper. Moreover, it is the first time that a semi-supervised learning with GAN is employed for the end to end task in autonomous driving.
2. Applying semi-supervised classification techniques to regression comes with the price of converting continuous labels of the dataset to a limited number of classes. This conversion will add the quantization error to our training, and also determining the number of classes for each application is nontrivial. Our proposed approach avoids this quantization error and reduces one hyper-parameter.
3. Our approach generates high quality images, smaller label prediction error and more stable training compared with the state-of-the-art Improved GAN technique (Salimans et al., 2016).

The remainder of the paper is organized as follows. In the next section, related works in the literature will be reviewed. Then, some preliminary background on generative adversarial networks will be described briefly. The proposed method will be presented in section 4. In section 5, the results of the experiments will be depicted. Finally, section 6 will conclude the paper.

## 2 RELATED WORK

In this section, we will review some relevant work to the idea of this paper in generative adversarial network and semi-supervised learning.

Deep learning has shown great successes in various domains such as natural language processing (NLP), autonomous driving, gaming, and generative models. One of the most recent achievements is generative adversarial network (GAN) (Goodfellow et al., 2014), which is well known because of generating synthetic realistic images. GAN corresponds to a minimax two-player game where two deep networks are trained simultaneously: a generative model  $G$  that captures the data distribution, and a discriminative model  $D$ , which computes the probability of a sample coming from the training data rather than the generator. Although GANs have shown a great success in generating sharp looking and fairly realistic images, they still encounter different critical issues such as stability of training (Arjovsky & Bottou, 2017; Warde-Farley & Bengio, 2017), mode dropping, evaluating GANs (Theis et al., 2015) and labeling generated samples. Hence, to address these issues, different types of GANs have been proposed in the literature such as DCGAN (Radford et al., 2015), WGAN (Arjovsky et al., 2017), least-square GAN (Mao et al., 2016), and Conditional GAN (Gauthier, 2014).

Semi-supervised techniques based on deep generative networks target improving the supervised task by learning from both labeled and unlabeled samples (Kingma et al., 2014). Using semi-supervised learning would be beneficial when labeled samples are not easy to obtain and we have a small set of labeled samples and more number of unlabeled data. Using deep generative models (and generative adversarial networks recently) in semi-supervised learning has introduced remarkable improvements to the field (Kingma et al., 2014; Salimans et al., 2016; Odena, 2016; Nguyen et al., 2016; Springenberg, 2015; Tachibana et al., 2016; Denton et al., 2016; Rasmus et al., 2015; Li et al., 2017). For example, the Improved-GAN technique shows competitive test errors and high quality generated samples over MNIST, CIFAR-10, SVHN and ImageNet datasets. In contrast to Improved-GAN, Triple-GAN (Li et al., 2017) employs two separate networks for classification and discrimination, which has more parameters to learn, and training the networks would be more difficult. However, almost all of these techniques target the classification problem and if we want to apply them to a regression problem, we need to classify (or quantize) the continuous labels to a limited number of classes. On the other hand, there are a couple of semi-supervised regression methods in the literature (Zhou & Li, 2005; Yang et al., 2016), but they do not use the power of deep learning based generative models such as GAN.

## 3 GENERATIVE ADVERSARIAL NETWORKS

Generative adversarial networks include two separate deep networks: the generator and discriminator. The generator takes in a random variable,  $z$ , with the distribution  $p_z(z)$  and maps it to the data distribution  $P_{data}(x)$ . The output distribution of the generator should converge to the data distribution during the training. On the other hand, the discriminator is expected to discern real samples from generated samples by giving the output of 1 or 0 respectively. In the GAN training process, the generator and discriminator are used to generate samples and classify them respectively by improving the performance of each other in an adversarial manner. In this regard, an adversarial loss function is employed in training the generator and discriminator (Goodfellow et al., 2014):

$$\min_G \max_D E_{x \sim P_{data}(x)} [\log D(x)] + E_{z \sim P_z(z)} [\log(1 - D(G(z)))] \quad (1)$$

This is a two-player minimax game for which the Nash-equilibrium point should be derived. Finding the solution of this game is non-trivial and there has been a huge volume of research in this domain (Shrivastava et al., 2016; Sixt et al., 2016; Springenberg, 2015; Li et al., 2017; Salimans et al., 2016).

The original GAN technique is not able to predict the label of the generated samples. The beauty of the Improved-GAN method (Salimans et al., 2016) is to combine the task of classification and discrimination into the discriminator network (i.e. using one network for performing the two tasks). Improved-GAN modifies the architecture of the discriminator to have  $N+1$  outputs, where  $N$  represents the number of classes in the training dataset (see the Fig. 1). The first  $N$  outputs should predict the

probability of an input to belong to each class,  $p(y|x, y < N + 1)$ ; and the last output represents the probability of the sample to be fake  $p(y = N + 1|x)$ . Then Improved-GAN tries to maximize the probability of predicting correct labels over the real and generated data as follows:

$$\max E_{x,y \sim P_{data}(x,y)}[\log p_{data}(y|x)] + E_{x \sim G}[\log p_{data}(y = K + 1|x)]. \tag{2}$$

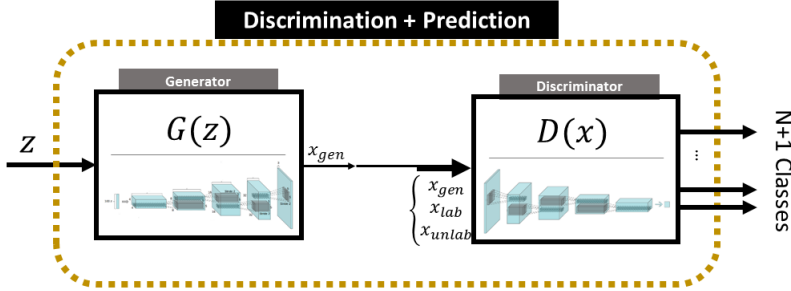


Figure 1: A simplified schematic of the original Improved-GAN technique

Moreover, Improved-GAN uses the feature matching technique to address the instability issue of the generator. In contrast to traditional GAN techniques which try to maximize the output of the discriminator for generated samples, feature matching tries to maximize the matching between the statistics of the generated and real samples inside the discriminator:

$$L_{feature\_matching} = ||E_{x \sim p_{data}} f(x) - E_{z \sim p_z(z)} f(G(z))|| \tag{3}$$

where  $f(x)$  represents the output of an activation function of an intermediate layer of the discriminator.

#### 4 METHODOLOGY

The focus of this research is on semi-supervised learning based on generative adversarial networks to solve the regression problem. We intend to apply GAN to generate realistic and high quality samples, as well as predicting the continuous labels corresponding to those generated samples. The core idea of our work is inspired by the Improved-GAN technique (Salimans et al., 2016), and we try to extend Improved-GAN to be able to cover regression as well.

Our proposed method, Reg-GAN, is comprised of a generator, which is responsible for generating realistic samples close to the content of the training dataset, and a discriminator, which is responsible for both validating the generated samples and predicting continuous labels of these samples. The generator is trained by employing the feature matching loss technique which is explained in the previous section. It is worth mentioning that the feature matching loss is the average of the absolute difference between the output of an intermediate layer of the discriminator for the real and generated samples.

We propose two architectures for the discriminator in our GAN (see Figs. 2 and 3) to address the semi-supervised regression with generative adversarial networks. In the first approach, the discriminator is built with two outputs: one is responsible for predicting the label, and the other predicts the probability that the generated sample is real/fake. If we assume that the labels can be mapped (or normalized) to the range of  $[0, 1]$ , then we can use a sigmoid nonlinearity in the last layer of the discriminator network. The discriminator is trained by using the combination of the usual unsupervised GAN loss function and a supervised regression loss:

$$\begin{aligned} \text{Loss}_D &= L_{\text{unsupervised}} + L_{\text{supervised}} \\ L_{\text{supervised}} &= ||y - \hat{y}|| \\ L_{\text{unsupervised}} &= E_{x \sim P_{data}(x)}[(1 - D(x))^2] + E_{z \sim P_z(z)}[D(G(z))^2] \end{aligned} \tag{4}$$

where  $z$  represents the noise drawn from a uniform or normal distributions.  $x$  and  $G(z)$  describe the true and generated images respectively. The term  $y$  refers to the true value of the label and  $\hat{y}$  indicates the predicted labels. It is worth mentioning that, we employ least-square loss functions (Mao et al., 2016) in the unsupervised part of the equation. In addition, the supervised regression error (i.e. the difference between the predicted and the true labels) is added to the discriminator loss function which helps to generate labels for the unseen or generated samples. The block diagram of the proposed method is shown in Fig. 2.

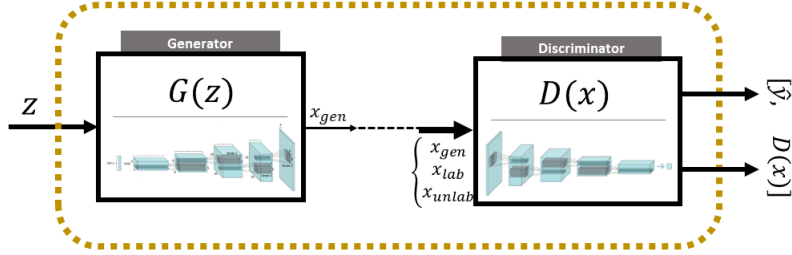


Figure 2: Architecture 1: the proposed semi-supervised regression with GAN (Reg-GAN) where both the  $D(x)$  and predicted labels are generated from the deep convolutional neural network.  $x_{gen}$ ,  $x_{lab}$ , and  $x_{unlab}$  represent the unlabeled generated, labeled real and unlabeled real samples respectively.

In the second approach (Fig. 3), instead of having two outputs in the discriminator, we keep only the regression output from the deep convolutional neural network to predict labels. Then we feed the labels to another function to assign an index to the generated samples based on the predicted label from the preceding convolutional neural network. In other words, instead of differentiating true and generated samples by the network directly, we can employ a separate kernel function (Eq. 5) on the regression output for deciding whether predicted labels are realistic or not. The kernel function is responsible to assign an index to each input label. If the predicted label is within the normalized range of true labels (i.e. between 0 and 1), then the assigned index is 1 and otherwise, the index will be assigned exponentially by a number less than 1 according to the distance of the predicted value from the target range of true labels. The training procedure of the proposed approaches are briefly portrayed in Algorithm 1.

$$\text{Kernel Function } K(\hat{y}) = \begin{cases} \exp(\hat{y}), & 0 \leq \hat{y} \\ 1, & 0 < \hat{y} \leq 1 \\ \exp((1 - \hat{y})), & 1 < \hat{y} \end{cases} \quad (5)$$

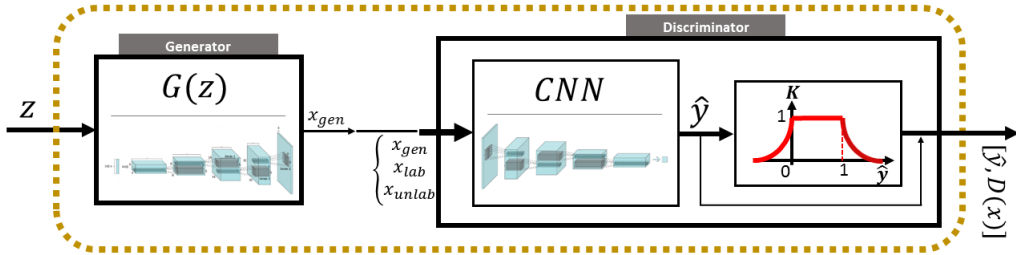


Figure 3: Architecture 2: the proposed semi-supervised regression with GAN (Reg-GAN) where only the labels are predicted by the deep convolutional neural network (CNN).  $x_{gen}$ ,  $x_{lab}$ , and  $x_{unlab}$  represent the unlabeled generated, labeled real and unlabeled real samples respectively.

---

**Algorithm 1** Semi-supervised regression with GAN. We use default values for  $\alpha = 0.0005$ ,  $\beta=0.5$

---

**Require:** The Adam hyperparameters  $\alpha$ ,  $\beta$ , the number of batches  $m$

**Require:** Initial discriminator parameters  $w_0$  and initial generator parameters  $\theta_0$

```

1: for  $\theta$  has not converged do
2:   for  $i = 1, \dots, m$  do
3:     Sample real data  $x, y \sim P_{data}(x, y), z \sim P_z(z)$ 
4:      $L_D^{(i)} \leftarrow E_{x \sim P_{data}(x)}[(1 - D(x))^2] + E_{z \sim P_z(z)}[D(G(z))^2] + \|y - \hat{y}\|$ 
5:      $w \leftarrow Adam(L_D^{(i)}, w, \alpha, \beta)$ 
6:      $L_G^{(i)} \leftarrow L_{feature\_matching}$ 
7:      $\theta \leftarrow Adam(L_G^{(i)}, \theta, \alpha, \beta)$ 
   end for
end for

```

---

## 5 EXPERIMENTS AND RESULTS

In this section, our method will be evaluated through some experiments from different point of views such as regression prediction error, quality of generated samples and stability of training. The main objective of the experiments is to show that our proposed architectures are able to learn data generation and label prediction even in the case of having limited data and among them only a small amount of samples are labeled.

### 5.1 DATA AND EXPERIMENTAL SETUP

For our experiments, we use a publicly available driving dataset <sup>1</sup>. The dataset contains images taken from a front facing camera mounted on the car with their corresponding steering angles as labels. We randomly choose 7200 samples from the dataset for training and 9000 samples for test. We aim at evaluating our technique when the number of available samples and the number of labeled samples are small. Hence, we did not incorporate more samples from the test set into our training data. The label of the samples falls within the range of  $[-2.79, 8.75]$  which are further normalized into the range of  $[0, 1]$  in a linear way. We use the average normalized prediction error over the test set to compute the test error as following:

$$\text{test\_error} = \frac{1}{N} \sum_{j=1}^N \frac{\|\hat{y}_j - y_j\|}{\|y_{max} - y_{min}\|} \times 100 \quad (6)$$

where  $N$  is the number of test samples, and  $y_{min}$  and  $y_{max}$  represents the minimum and maximum value of the groundtruth labels (i.e. 0 and 1 respectively). We use the available Improved-GAN code, which is written using 'Theano' python library and 'Lasagne' deep learning library, as a baseline to implement our proposed methods. We perform experiments for 800 iterations with a learning rate of  $\alpha=0.0005$  for our methods and 0.0003 for the original improved GAN (Salimans et al., 2016). The experiments are run on a single NVIDIA Tesla P100 GPUs.

### 5.2 EXPERIMENTAL RESULTS

For training our proposed architectures, we used different scenarios with different number of labeled samples in a semi-supervised learning setting. We perform the training by considering 1000, 2000, 4000, and "All" labeled samples. Moreover, for each scenario, we also feed all the training set without labels as unlabeled samples to the algorithm. We compare our proposed architectures with the state-of-the-art Improved-GAN semi-supervised learning approach (Salimans et al., 2016). We chose this method over other similar techniques because it outperforms them. In order to fit our dataset into the Improved-GAN classification framework, we discretized the normalized continuous labels into 10 number of classes (we assign labels in the range of  $[0, 0.1) \rightarrow 0$ , and  $[0.1, 0.2) \rightarrow 1$ , and ...  $[0.9, 1] \rightarrow 9$ ). Bear in mind that this discretization will add some unavoidable quantization error to our training.

---

<sup>1</sup>The dataset can be downloaded from: <https://drive.google.com/file/d/0B-KJCaaF7elleG1RbzVPZWV4Tlk/view>

Table 1: Test errors using 1000, 2000, 4000 and All labeled samples

Model	1000	2000	4000	All
Improved_GAN	4.38%	4.22%	4.07%	4.06%
Reg-GAN (Architecture 1)	2.43%	2.40%	2.39%	2.36%
Reg-GAN (Architecture 2)	3.81%	3.58%	2.23%	2.21%

The results of the experiments are depicted in Table 1. From Table 1, we can note that our proposed architectures outperform the Improved-GAN approach in all scenarios significantly. Our method, Reg-GAN, gives the average improvement of 42.7% and 29.7% over the traditional improved-GAN approach for the architecture 1 and 2 respectively. An example of the generated samples from different techniques is shown in Fig. 4. These samples are derived after training the networks over 1000 samples.

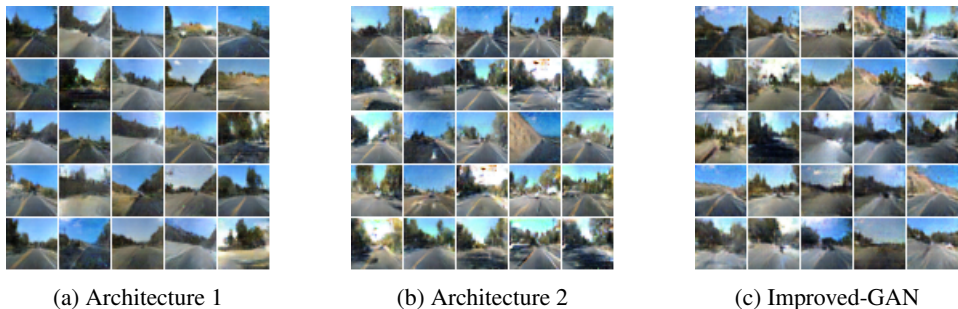


Figure 4: Sample generated images by using (a) Architecture 1, (b) Architecture 2, and (c) Improved-GAN when 1000 labeled samples are used for training.

Furthermore, to assess the training stability of the methods, we plot the training and test errors in Fig. 5. The plots are given for 400 iterations -rather than 800- to be able to track the variations of each plot more clearly. Fig. 5 shows that some sharp peaks occur while training Improved-GAN which can affect the stability of training. On the other hand, we can note that our proposed approaches (see Fig. 5-(a) and (b)) are performing more stable and smoother than the original Improved GAN.

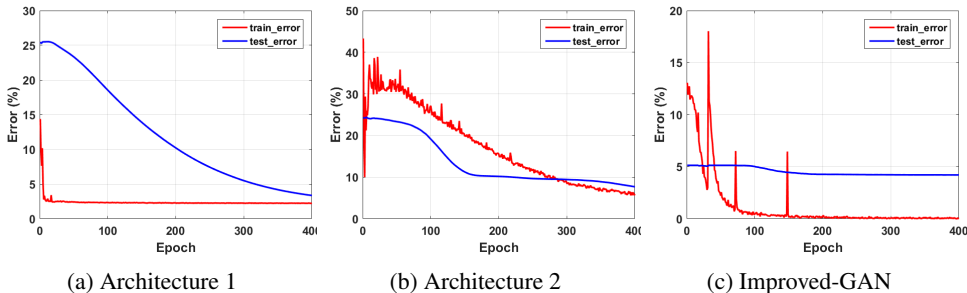


Figure 5: Train and test errors are compared over 400 iterations using (a) Architecture 1, (b) Architecture 2, and (c) Improved-GAN when 2000 labeled images are used

## 6 CONCLUSION AND FUTURE WORK

This work concerned solving the semi-supervised regression task by incorporating generative adversarial networks. The conventional semi-supervised learning with GAN are suitable for classification task that is using them for the regression task requires to convert continuous labels to a limited number of classes. This conversion will add the quantization error to the training, and also determining the number of classes for each application is non-trivial. This work proposes a semi-supervised regression

task using GAN which overcomes the above mentioned problems that arise using semi-supervised classification techniques to solve the regression task. We did experiments on a publicly available driving dataset where continuous steering angles were used as the labels with the corresponding images. We showed that our proposed approaches outperform the state-of-the-art Improved-GAN technique in the literature. We summarize our plan for future work in the following:

1. The idea of this work can be extended to cover classification problems as well by assigning the regression label output to predict the class labels. However, the performance of this approach on the classification problems needs to be investigated.
2. The idea of semi-supervised regression may have other applications such as face detection, and apparent age estimation from a single image. Our method can be evaluated on those applications as well.

## REFERENCES

- Martin Arjovsky and Léon Bottou. Towards principled methods for training generative adversarial networks. In *NIPS 2016 Workshop on Adversarial Training. In review for ICLR*, volume 2016, 2017.
- Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.
- Ershad Banijamali, Ali Ghodsi, and Pascal Poupart. Generative mixture of networks. *arXiv preprint arXiv:1702.03307*, 2017.
- Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Praseon Goyal, Lawrence D Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, et al. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016.
- Emily Denton, Sam Gross, and Rob Fergus. Semi-supervised learning with context-conditional generative adversarial networks. *arXiv preprint arXiv:1611.06430*, 2016.
- Jon Gauthier. Conditional generative adversarial nets for convolutional face generation. *Class Project for Stanford CS231N: Convolutional Neural Networks for Visual Recognition, Winter semester*, 2014(5):2, 2014.
- Arna Ghosh, Biswarup Bhattacharya, and Somnath Basu Roy Chowdhury. Sad-gan: Synthetic autonomous driving using generative adversarial networks. *arXiv preprint arXiv:1611.08788*, 2016.
- Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.
- Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.
- Brody Huval, Tao Wang, Sameep Tandon, Jeff Kiske, Will Song, Joel Pazhayampallil, Mykhaylo Andriluka, Pranav Rajpurkar, Toki Migimatsu, Royce Cheng-Yue, et al. An empirical evaluation of deep learning on highway driving. *arXiv preprint arXiv:1504.01716*, 2015.
- Matthew Johnson-Roberson, Charles Barto, Rounak Mehta, Sharath Nittur Sridhar, and Ram Vasudevan. Driving in the matrix: Can virtual worlds replace human-generated annotations for real world tasks? *arXiv preprint arXiv:1610.01983*, 2016.
- Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- Diederik P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems*, pp. 3581–3589, 2014.



- Alex Kuefler, Jeremy Morton, Tim Wheeler, and Mykel Kochenderfer. Imitating driver behavior with generative adversarial networks. *arXiv preprint arXiv:1701.06699*, 2017.
- Chongxuan Li, Kun Xu, Jun Zhu, and Bo Zhang. Triple generative adversarial nets. *arXiv preprint arXiv:1703.02291*, 2017.
- Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. *arXiv preprint ArXiv:1611.04076*, 2016.
- Tan Nguyen, Wanjia Liu, Ethan Perez, Richard G Baraniuk, and Ankit B Patel. Semi-supervised learning with the deep rendering mixture model. *arXiv preprint arXiv:1612.01942*, 2016.
- Augustus Odena. Semi-supervised learning with generative adversarial networks. *arXiv preprint arXiv:1606.01583*, 2016.
- Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.
- Antti Rasmus, Mathias Berglund, Mikko Honkala, Harri Valpola, and Tapani Raiko. Semi-supervised learning with ladder networks. In *Advances in Neural Information Processing Systems*, pp. 3546–3554, 2015.
- Ruslan Salakhutdinov, Andriy Mnih, and Geoffrey Hinton. Restricted boltzmann machines for collaborative filtering. In *Proceedings of the 24th international conference on Machine learning*, pp. 791–798. ACM, 2007.
- Tim Salimans, Ian J. Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *CoRR*, abs/1606.03498, 2016. URL <http://arxiv.org/abs/1606.03498>.
- Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Josh Susskind, Wenda Wang, and Russ Webb. Learning from simulated and unsupervised images through adversarial training. *arXiv preprint arXiv:1612.07828*, 2016.
- Leon Sixt, Benjamin Wild, and Tim Landgraf. Rendergan: Generating realistic labeled data. *arXiv preprint arXiv:1611.01331*, 2016.
- Jost Tobias Springenberg. Unsupervised and semi-supervised learning with categorical generative adversarial networks. *arXiv preprint arXiv:1511.06390*, 2015.
- Ryosuke Tachibana, Takashi Matsubara, and Kuniaki Uehara. Semi-supervised learning using adversarial networks. In *Computer and Information Science (ICIS), 2016 IEEE/ACIS 15th International Conference on*, pp. 1–6. IEEE, 2016.
- Lucas Theis, Aäron van den Oord, and Matthias Bethge. A note on the evaluation of generative models. *arXiv preprint arXiv:1511.01844*, 2015.
- D Warde-Farley and Y Bengio. Improving generative adversarial networks with denoising feature matching. *ICLR submissions*, 8, 2017.
- Yi Yang, Jiping Liu, Shenghua Xu, and Yangyang Zhao. An extended semi-supervised regression approach with co-training and geographical weighted regression: A case study of housing prices in beijing. *ISPRS International Journal of Geo-Information*, 5(1):4, 2016.
- Zhi-Hua Zhou and Ming Li. Semi-supervised regression with co-training. In *IJCAI*, volume 5, pp. 908–913, 2005.