AUTISM ASSISTANCE USING EMOTION-CONDITIONED GAN FOR FACIAL EXPRESSION

By

Daniela Nogueira

Thesis submitted in partial compliance of the FT MSc Data Science Program requirements for the academic year 2022/23, Department of Computing, Goldsmiths, University of London.

Supervisor: Dr. Sean McGrath

Abstract

As a rapidly developing field of science, image processing is currently one of the most active areas of study in the medical industry. Emotion analysis, market research, the prediction of the prevalence of neurological disorders, psychological issues, and so on all make use of facial expressions. Since then, academics have begun to pay more attention to it. Facial expression processing and recognition play crucial roles in human social interaction. Previous studies have shown that people on the autism spectrum may have trouble understanding emotions conveyed by others' faces. It's been found that Generative Adversarial Networks (GANs) are effective deep generative learning architectures. Using GANs, we have been able to produce synthetic images of human faces, landscapes, and buildings with a level of realism never seen before. GANs can be used for more than only generating images; they can also be used to alter existing images. Assistive technology is one of the greatest innovations in recent times to aid persons with autism in leading more fulfilling lives. The system proposed in this study aims to use Generative deep learning models to generate 8 facial emotions: neutral, contempt, anger, disgust, fear, happiness, sadness, contempt, and surprise, in order to teach emotions to autistic children, who have significant difficulties recognizing and responding to the emotional and mental states of others.

Keywords: image processing; facial expression processing and recognition; generative adversarial networks; neurological disorders; autism spectrum; social interaction; assistive technology; autistic children

Acknowledge

I would like to take this opportunity to thank all the people who have supported me throughout my thesis.

Firstly, I want to express my gratitude to my supervisor, Sean McGrath, for his patience, hard work and guidance throughout the thesis.

I want to express my gratitude to all of my professors who helped and taught me so diligently during this master's year. All the knowledge I gained from them had a significant impact on the development of this thesis.

I would like to thank my family for their enduring support and love.

I want to express my gratitude to my colleagues for their unconditional support and for taking on the unenviable task of keeping me confident throughout the thesis.

Lastly, I would like to thank my late dad, Aurélio, whose examples of strength and resilience have helped me unendingly, and to whom this thesis is dedicated.

Table of Contents

List of Abbreviations

Assistive Technology **(AT)**

Autism Spectrum Disorder **(ASD)**

Action Unit **(AU)**

Centers for Disease Control and Prevention **(CDC)**

Conditional Generative Adversarial Networks **(CGANs)**

Convolutional Networks **(CNNs)**

Deep Convolutional Generative Adversarial Networks **(DCGANs)**

Diagnostic and Statistical Manual of Mental Disorders **(DSM-5)**

Facial Action Coding System **(FACS)**

Generative Adversarial Networks **(GANs)**

National Institute on Deafness and Other Communication Disorders **(NIDCD)**

Research and Development **(R&D)**

List of Appendices

Chapter 1

1. Introduction

1.1.Autism Spectrum Disorder

According to data provided by the Centers for Disease Control and Prevention (CDC) and Autism Spectrum Disorder 2018 Estimates, an autism diagnosis is given to approximately one in every 44 children [1]. Although Leo Kanner was the one who made the initial discovery of it in 1943 [2] our current understanding of ASD has made enormous advancements in terms of diagnosis and treatment in recent decades. Based on most of the scientific opinion, this neurological and developmental disorder arises from the combination of hereditary and environmental variables. This indicates that a child can have a susceptibility toward developing autism from the moment they are born, and that vulnerability can subsequently be triggered by physical factors in their surroundings. It can manifest itself in a broad range of ways and to varying degrees throughout a person's lifetime. This condition is also commonly referred to by its acronym, ASD. According to the National Institute of Mental Health, autism can be diagnosed at any age; however, symptoms often develop within the first two years of a person's existence [3]. Children with ASD can learn to function independently while attending mainstream schools if they receive the appropriate early intervention. This can considerably enhance the quality of life for the vast majority of them.

There is a widespread misunderstanding that children who have autism do not experience any feelings. They have feelings, just like any other person, but express those emotions in a different way [4]. According to the Diagnostic and Statistical Manual of Mental Disorders (DSM-5), a guide created by the American Psychiatric Association that is used by medical professionals to diagnose mental disorders, people with autism experience difficulties such as troubles in communicating effectively, recognizing emotions, and problems in interacting socially. These issues can vary from an uncommon social approach and failure in normal discussions all the way up to a complete absence of initiation into social engagement [5]. Autistic children can distinguish between different emotions, but they need more prompting to choose the correct label, according to the findings of a study that looked at a teaching technique for emotions that was published in 2015 [6]. They gained experience classifying different emotions by using photographs and a variety of scenarios. Participants were, for example, asked to match the appropriate feeling card with the appropriate setting card (such as a picture of a girl receiving a Christmas present being paired with a smiling face). According to the study's findings, autistic children can recognize emotions when given the appropriate amount of instruction and repetition. Children with autism disorder who are able to talk will often say things that are either meaningless or do not relate to the conversations they are having with others, as stated by the National Institute on Deafness and Other Communication Disorders (NIDCD). In addition to this, the children have trouble comprehending the viewpoint of another person and are unable to anticipate or comprehend the activities of other people [7]. In addition, they regularly avoid making eye contact, which can give the impression that they are unfriendly or apathetic. This impression might be created since it gives the appearance that they are avoiding you. They also struggle with not having the capacity to employ meaningful gestures or other non-verbal skills that can complement their speech. Because of this, they end up feeling an incredible amount of frustration. As a means of venting their anger, they may either vocally explode or indulge in other sorts of undesirable behavior.

1.2.The significance of assistive technologies for children with autism spectrum disorder

Children with ASD are frequently uninterested in facial expressions [8]. To have empathy for another person, which is described as the ability to understand the sentiments of another, requires you to place yourself in the position of the other person and to comprehend the reasons why they may be experiencing the emotions that they do, without necessarily having those feelings yourself. On the other hand, sympathy is defined as the ability to understand another person by experiencing the same feelings that they do from your own perspective. People on the autism spectrum may not have the capacity for either sympathy or empathy [9]. When someone gets seriously hurt, they might show signs of glee, but it's also possible that autistics have no reaction at all. It's likely that they struggle with social semiotics, which is the competence of an individual to behave appropriately in a certain social scenario. As a consequence of this, it may be challenging for the individual to develop healthy relationships with others.

Because people with autism may not appear to have any obvious physical disabilities, it is easy to forget how valuable assistive technology (AT) may be for them, in the way that it helps those people enhance the overall quality of their lives and, as a result, their well-being. Through the use of instructional activity and meditation, technology can be utilized to support visually the process of emotional regulation. Children who have ASD might benefit greatly from improving their ability to regulate their emotions by seeing how others behave and being given visual signals based on the context or circumstance they are in. In general, disabled people's ability to perform daily tasks can be enhanced by the use of several types of assistive technology in a wide range of areas, including communication (speaking, typing, writing), literacy (reading, learning, and walking), mobility, and more. AT for people with autism of all ages has been developed, and research and development (R&D) efforts in this area have increased in recent years. Auxiliary aids in inclusive learning environments, such as schools, may become more common if these people regularly use technology. The majority of experts agree that picking the right assistive technology for people with autism should be done carefully, taking into account the severity of their disease [10]. Each individual with ASD has a different set of signs and symptoms. This means that not everyone will benefit equally from the same pieces of assistive technology. Certified experts are the only ones who can detect the difference and provide the necessary help here.

According to the World Health Organization, more than 2 billion individuals will require at least one assistive product by 2030 due to an ageing global population and an increase in noncommunicable diseases, with many older people requiring two or more [11]. Children who learn to use their hearing aids well have a much greater chance of developing strong language abilities, which opens many doors for them in school and the workforce. People are often excluded, isolated, and locked into poverty when assistive technology is not available to them, which increases the impact that disease and disability has on an individual, their family, and society as a whole.

The organization Autism Speaks claims that 31% of children with ASD have an intellectual handicap (intelligence quotient [IQ] < 70]. By some estimates, up to 40% of people with autism are non-verbal [12]. AssistiveWare's Proloquo2Go, which contains more than 10 000 different words, is simple to adapt to one's specific physical or mental requirements and may be utilized in a variety of languages. It is an app for augmentative communication and speech therapy. This app was not created specifically for people with autism, but it is extremely useful and cost-effective for those who cannot express themselves effectively, helping them to develop speech and language skills. Many specialist companies have created cards and games to help with such skills. There are games such as Ladders designed to reinforce empathy, and other feelings-focused Uno Cards that aim to reinforce social communication skills. Caregivers of autistic children who have difficulty with motor coordination have reported beneficial outcomes from their children's usage of technology with touch screens. A child can use a smartphone app to type in the words and then use their finger to swipe through the many possibilities, for example. Because of this, the child won't need a pencil or to try to find cards with specific images on them. A rise in the number of resources that are available and the opportunity to make use of them increase one's feeling of identity and self-esteem.

1.3.Proposed Assistive Tool: Conditional DCGAN for autistic children

In this project, we present a GAN proposal that generates facial expressions that are conditioned on the following eight emotions: neutral, happiness, sadness, anger, surprise, contempt, disgust, and fear. In addition, the purpose of the model is to teach autistic children how to learn and identify the emotions of others by generating facial expressions. Due to the fact that the model has been conditioned, the children are able to select the specific type of emotion they wish to observe at a given moment, and they can do so while having access to a large variety of images of different faces that can be compared to one another. Because it is still a work in progress and there are many steps left to do before this project can become a more interactive assistive technology, we will begin by providing a base, a prototype for a more finished and comprehensive product that will be developed in the future with the access to more and better resources, time, and an even deeper understanding of the topic at hand. Numerous projects in the field of data science have been developed to assist autistic children, but have primarily focused on detecting or predicting facial expressions on autistic children's faces, and very few projects exist in the sense of producing these emotions so that autistic children can, with the help of an assistive technology, explore and produce the visuals and feelings on their own.

The results of our model, which we call Conditional DCGAN, are stable, which means that the model does not collapse and instead generates pictures that are distinct from one another. We were also successful in developing a model that successfully struck a healthy balance between the generator and the discriminator. This model assigned the discriminator the challenging task of distinguishing real images from fake ones, while the generator was tasked with creating ever more realistic representations of the facial expressions.

Chapter 2

2. Literature review

An individual's ability to communicate a variety of social signals, such as indications of where the person's attention is directed, is made possible by the highly movable features of the human face. However, the facial expressions of emotion are the ones that have received the most attention from researchers since they are thought to be an important source of information about a person's current emotional state (Ekman, 1992) [32]. In 1862, Duchenne was the first person to conduct a scientific study on the facial expression of emotion. During this study, he discovered that it was possible to stimulate the independent movement of individual facial muscles on a participant who had lost the ability to move the muscles in his face due to nerve damage [33]. In the wake of this, Darwin (1872) proposed that certain facial displays of emotion were generally discernible. This means that facial expressions of emotion are universally acknowledged and easily comprehended by specific facial motions throughout all countries of the world [34]. Ekman et al. (1992) elaborated on this by suggesting that "basic facial expressions of emotion" such as happiness, surprise, fear, contempt, anger, and sadness are universally conveyed using the same mix of facial movements no matter where in the world one is located [35]. There are, however, counterexamples to this idea. For instance, Russell (1994) raised doubts about the veracity of the emotion idea as a whole, drawing the conclusion that different cultures tend to assign different meanings to similar facial expressions [36]. Nelson (1987) observes that even by the age of five, the ability to correctly identify a range of emotional facial expressions is only rudimentary [37]. Emotional face perception is assumed to include a series of brain processes that mature from childhood into maturity (Batty and Taylor, 2006) [38].

A facial detection system that not only identifies faces in a picture but also calculates the type of emotions based on facial features has been the subject of research and development by scientists and professionals in the field of computer science for quite some time. One of the earliest studies on automatic facial detection was conducted by Bledsoe for the United States Department of Defense in 1960. Kanade [13] created the first totally automated and fully functional facial recognition system. By differentiating between computer-generated and human-extracted features, this method was able to quantify sixteen distinct facial characteristics. It did so by gauging the degree to which two sets of attributes diverged from one another. However, this method's accuracy can't improve above 75% at the most. Researchers found that showing autistic youngsters images of facial expressions helped them understand and process those emotions [14]. An human-computer interface [15] was developed in a project called "AURORA," which involved the use of a robot and allowed interaction between the child and the robot. To help teach special needs children about different emotions, another study [16] advocates using human-computer interaction to show the youngsters a series of short movies illustrating each emotion. Despite the wealth of research on the topic, there are still obstacles to overcome when working with autistic children to teach them about emotions. Facial expression is a visual representation of facial movements that is useful for communicating and interpreting human behavior. As a result of their impressive generating capability, Generative Adversarial Nets have recently become popular for creating fictitious images. Models of GANs that take into account conditions and limitations are the subject of active study. Previous research has investigated the effects of mixing many circumstances, including textual descriptions [17,18] and information about classes [19]. The approaches of image super-resolution [20], future frame prediction [21], image in-painting [22], image-to-image translation [23], and multi-target domain transfer [24] are of particular interest for this work since they explore image-based conditioning. Thus, numerous conditional GAN-based techniques have been the subject of intensive research for the purpose of achieving these goals. Deep convolutional GANs (DCGANs) were initially described for the purpose of unsupervised image production by Radford et al. [25]. This approach has the ability to learn hierarchical representations of data and produce facial animations that are believable. The ExprGAN was first proposed by Ding et al. [26] as a tool for manipulating facial expressions with adjustable expression intensity. Choi et al. [25] introduced a multi-domain image-to-image translation strategy called StarGAN. This strategy could also be employed in facial expression transfer, generating diverse faces in hair, expression, gender, and other categories. The goal of this strategy was to synthesize images across domains using a single model. To manipulate the intensity of individual Action Unit (AU) activations, Pumarola et al. [27] introduced an AU-based GAN model for synthetic facial animation. Following their work, Tripathy et al. [28] included posture factors in conditions for expressiveness and pose reenactment. To preserve the form of the face, Qiao et al. [29] and Kossaifi et al. [30] included geometric constraints in their generative networks. In order to achieve this, they first extracted the sparse landmark points of the target face and then converted them into a spatial heatmap.

Chapter 3

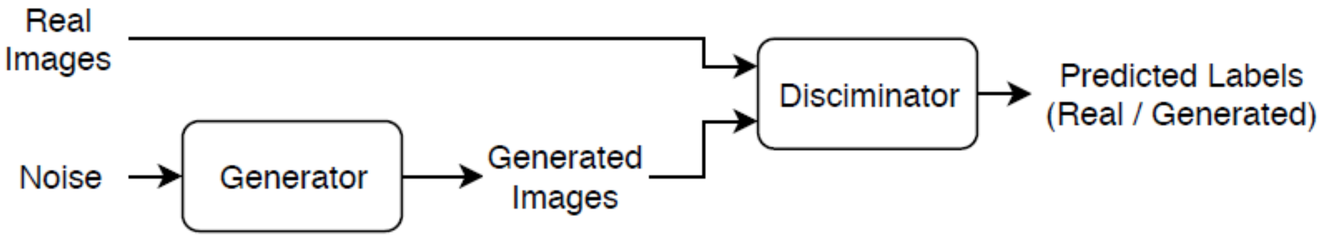
3. Methodology

3.1. Generative Adversarial Networks

GANs (Generative Adversarial Networks) are one of the most popular Deep Learning topics today, having been introduced in 2014 in a paper by Goodfellow at the University of Montreal. They are frequently used to create images, videos, and sounds. Yann Le Cun, Facebook's AI research director, called GAN training "the most exciting idea in the last 10 years in ML." It is made up of two neural networks that have been trained to compete against each other. To clarify, the generator (G(x)), the artist, takes a random vector as input (a random point in latent space) and decodes it into a synthetic image that is fairly realistic and statistically almost indistinguishable from the real ones. The generator's output, which consists of fake images, and the real images from the training set, are fed to the discriminator (D(x)), the artist critic, who learns to distinguish them. D(x) is the probability that the input is real. D(x) must be 1 if the input is from the training set and 0 if it is generated. During training, the generator attempts to fool the discriminator, gradually improving its ability to create images that appear real, while the discriminator constantly improves the generator's capability by setting a high bar of realism for the generator images. When the discriminator can no longer distinguish between real and fake images, the process has reached equilibrium. Let us examine this from a mathematical standpoint. The training process of the entire game can be described in the following Minimax Objective function:

$$\min \max V(D, G) = \mathrm{E}_{x\sim\rho_{data}(x)}[\log D\,(x)] + \mathrm{E}_{z\sim\rho_z(z)}\,[\log\,(1 - D(G(z)))]$$

Where x is the actual image; z is the image of input generator; $\mathrm{E}_{x\sim\rho}data_{(x)}$ and $\mathrm{E}z{\sim}\rho_{z}(z)$ are the expected functions. G converts the variable into the probability that the image generated by the convert is a real image. The ideal distribution should converge to the data distribution. D() calculates the likelihood that the given sample was extracted from the training data $\mathcal{X}$. The Generator's goal is to minimize $\log\,(1 - D(G(z)))$, which means that when $D(G(z))$ is high, D assumes that $G(z)$ is a real image, and thus $1 - D(G(z))$ has low values, which the Generator wants to reduce more and more. On the other hand, the discriminator intends to maximize $D(x)$ and $1 - D(G(z))$. As previously stated, the generator intends to confuse the discriminator so that it can distinguish between x and z. To put it simply, if one wins, the other loses. Your opponent wishes to maximize your actions, whereas you wish to minimize them.



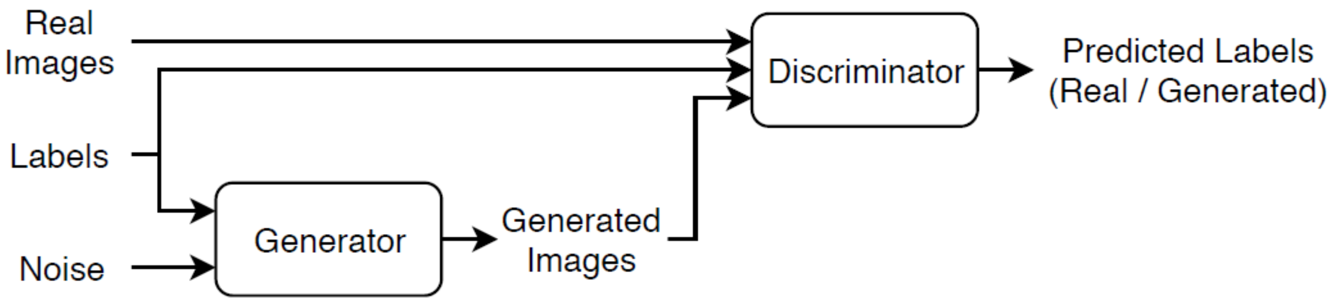
GAN architecture

Figure 1. Source from: 2022. [online] Available at: <https://www.mathworks.com/help/deeplearning/ug/train-conditional-generative-adversarial-network.html> [Accessed 20 September 2022].

This project presents a conditional version of generative adversarial networks first introduced in 2014 by University of Montreal PhD student Mehdi Mirza and Flickr AI architect Simon Osindero. This network can be easily built by conditioning both the generator and the discriminator with auxiliary information that could be, anything, such as class labels, which in our case, are the emotions: happiness, neutral, sadness, anger, surprise, disgust, contempt and fear. In its most basic form, GANs are an unsupervised generative modeling technique that generates synthetic data from provided data.

In its most basic form, GAN is an unsupervised generative modeling technique that generates synthetic data from given data. Conditional GANs (CGANs) convert generative modeling tasks to supervised learning tasks. This is due to the fact that during CGAN training, the generator learns to generate realistic examples of each label in the training data set, while the discriminator learns to distinguish between false and true examples. CGAN's discriminator does not learn to distinguish between classes. It simply learns to accept true compatible pairs while rejecting discordant pairs and pairs with incorrect examples. It is not enough for a CGAN generator to generate data that appears realistic to fool the discriminator. Generated examples must also match that label. Once the generator is fully trained, you can specify which samples CGAN to synthesize by passing the desired labels. To be more specific, the generator G(z, y) = x*|y in order to generate false examples x*|y, takes a random noise vector z and a label y as input. The discriminator is fed the real sample with its label (x,y) and the fake sample with the synthesis label (x*|y,y), in this sense, it returns a probability calculated by the sigmoid activation function (σ) indicating whether the input pair is real or fake.

Providing additional information is beneficial because it speeds up the convergence process and even random distributions followed by fake images show some pattern. You can also direct data generation and control the mode of synthesized data. In addition, as we will discuss further below, CGAN requires less human intervention in the modeling process than standard neural networks because we do not need to define craft loss for each specific task.



CGAN architecture

Figure 2. Source from: 2022. [online] Available at: <https://www.mathworks.com/help/deeplearning/ug/train-conditional-generative-adversarial-network.html> [Accessed 20 September 2022].

In this project, we will look at deep convolutional generative adversarial networks (DCGANs) with architectural constraints that we will investigate later. Given the use of convolutional networks (CNNs), also known as convnets, I believe it is necessary to provide a brief overview of what they are and how they work. The fundamental difference between a densely connected layer and a convolutional layer is that dense layers learn global patterns in their input feature space (for example, patterns involving all pixels in our case for face images), whereas convolutional layers learn local patterns: in the case of images, patterns found in small 2D windows of the inputs. The patterns they learn are translation invariant, which means that once learned, a convnet can recognize it anywhere, as opposed to a densely connected network, which would have to learn the pattern again if it appeared in a new location. Because the real world is fundamentally translation invariant, convnets learn representations with generalization power when processing images. Additionally, they learn spatial hierarchies of patterns, for example, the first convolutional layer will learn small local patterns like edges, the second convolutional layer will learn larger patterns made of the first layer's features, and so on. Convolutions operate on 3D tensors known as feature maps, which have two spatial axes (height and width) as well as a depth axis known as the channel axis. This output feature map is still a 3D tensor, but its depth is arbitrary because the output depth is a layer parameter, and the different channels in that depth axis no longer represent specific colors, but rather filters. Considering the CNN architecture, it's usually shaped with the aid of using 3 parts: convolutional layers, pool layers and completely linked layers. The first layers extract features, while the third maps the extracted features to the final output. Convolutional layers are made of numerous operations, along with convolution, a form of linear operation. Pixel values in images are saved in an array of numbers and a grid of small parameters referred to as kernels, an optimizable aid extractor is implemented to every picture location, making CNNs very powerful for image processing, because the features can seem everywhere within-side the picture, using optimization algorithms such as backpropagation and gradient descent.

3.2.Data Collection

In order to carry out experiments and proceed to the respective analyses and conclusions, this project used 2 different datasets, both collected from the Kaggle. The first dataset, CKPLUS, is available at https://www.kaggle.com/code/shawon10/ck-facial-expression-detection/data and is the extension of the Cohn-Kanade dataset developed by the team comprising Patrick Lucey and other participants [31]. The dataset includes 123 images of people's facial expressions between the ages of 18 and 50 from various genders and ethnicities, as well as 593 expression sequences and 951 image samples; however, the Kaggle version includes 981 samples. Each image has a pixel size of 3×48×48 and is labelled with one of 7 emotion categories: anger, contempt, disgust, fear, happiness, sadness, and surprise, as encoded by the Facial Action Coding System (FACS), which refers to a set of facial muscle movements that correspond to a displayed emotion. Originally invented by Carl-Herman Hjortsjö in 1970 and expanded by Paul Ekman and Wallace Friesen. The second dataset, FER2013, can be found at https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge/data and contains 35,887 examples of 48x48 pixel grayscale images of faces, each of which is classified into one of 7 labels: happiness, neutral, sadness, anger, surprise, disgust, and fear. Pierre-Luc Carrier and Aaron Courville prepared this dataset as part of an ongoing research project.

3.3.Procedure

During the development of this project, we go through six phases before reaching the final Conditional DCGAN, which is the project's proposal: Data Preprocessing, Unconditional GAN (Baseline), DCGAN Improvement 1, DCGAN Improvement 2, DCGAN Improvement 3 and, finally, Conditional DCGAN. We will go over each one in more detail shortly.

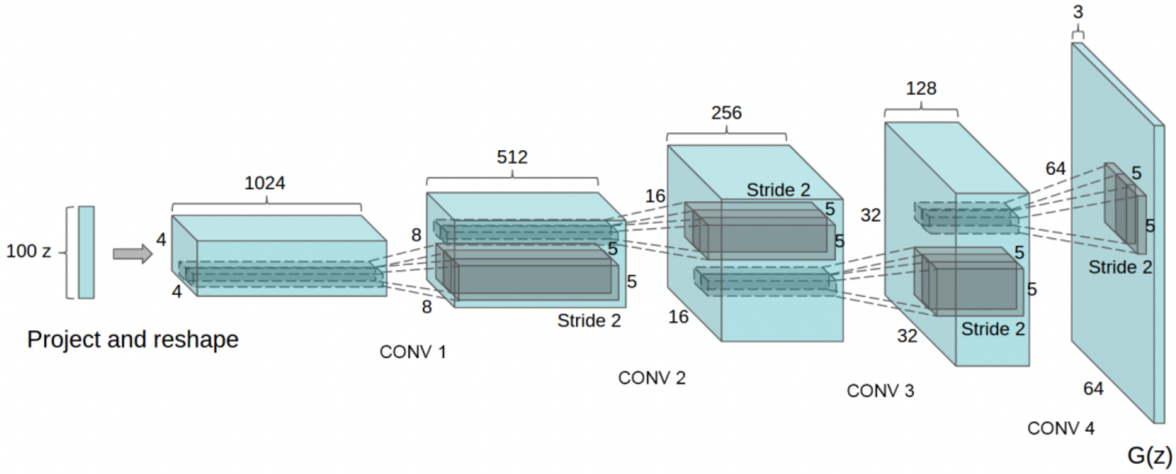


Figure 3: The DCGAN framework was used in the LSUN scene simulation [25]. A convolutional representation with a short spatial extent and several feature maps is projected from a 100-dimensional uniform distribution Z. Then, this high level representation is transformed into a 6x 64 pixel image using a sequence of four fractionally-strided convolutions. No pooling or fully connected layers are employed.

3.3.1.Data Preparation

During data preprocessing, for both datasets used in this work, we start by exporting, in grayscale mode, the images and their respective emotion labels from the directories where they are located, then we convert the data to a NumPy array and reshape each image array to (48,48,1) shape, after that, it is necessary to transform the data into floating points values and rescale them to a [-1,1] range, instead of values between 0 and 1, since the generator will use the hyperbolic tangent (tanh) activation function in the output layer which is computed as follows: $\frac{(e^x - e^{-x})}{(e^x + e^{-x})}$, where e is a mathematical constant that is the base of the natural logarithm. It is also worth noting that gradient descent works best when the data is homogeneous and constrained. Weights and bias are initialized with small values to prevent a few large inputs from dominating training. The final step in this phase is to concatenate the two arrays from different datasets to create a unique one.

3.3.2.Unconditional GAN (Baseline)

Beginning with the discriminator, we will develop a model that takes a candidate 48×48 grayscale image (real or synthetic) as input and classifies it into one of two binary classes: "generated image" (class=1) or "real image that comes from training set" (class=0). It is implemented a modest convolutional neural network that uses the Leaky ReLU activation function with a slope of 0.2 for all models, a 2×2 stride to downsample, and a Root Mean Squared Propagation (RMSProp) version of gradient descent as optimizer with a learning rate of 0.0008, a clipvalue of 1.0 and a decay of 0.00000001 to stabilize training. RMSprop uses a decaying average of partial gradients in the adaptation of the step size for each parameter. RMSprop adapts the step size for each parameter using a decaying average of partial gradients. The optimizer is the mechanism through which the network will uptake itself based on the data it sees. The network concludes with a single unit and sigmoid activation function, allowing the network to learn to predict values ranging from 0 to 1. As a regularization strategy, neural networks use a dropout layer. Doing so will randomly drop out or disable some of the layer's output features. The dropout rate is the percentage of features that are null. Usually, it ranges from 0.2 to 0.5, but in this case, we use 0.3. The idea is that by injecting noise into the output values of a layer, you can break up insignificant patterns that the network would start to memorize if the noise weren't there. Coming back to the activation function, it is preferable to use Leaky ReLUs instead of the usual ReLU function which works by truncating negative values to zero, for the DCGANs architecture, Leaky ReLUs help gradients to flow more easily through the architecture by allowing small negative values pass through the network, that is, the function calculates the largest value between the resources and a small factor. Leaky ReLUs also represent an attempt to solve the "Dead ReLU" problem. This happens when a neuron gets stuck in a state where the Relu unit always produces zero for all inputs. This is especially important for GANs as the only way the generator can learn is by getting the gradients from the discriminator. GAN training can be hindered by sparse gradients. Sparsity is a desirable property in most cases, but not in ours. Sparsity can be caused by two things: max pooling operations and ReLU activations, consequently, it is recommended to use stride in convolutional layers to perform downsampling in the discriminator model, and fractional step in deconvolutional layers for upsampling in the generator. In summary, downsampling is used to reduce the number of feature map coefficients to be processed, as well as to induce a hierarchy of spatial filters, allowing successive convolutional layers to observe increasingly larger windows according to the proportion of the original input that cover. Finally, we use a Flatten layer to reduce the multidimensional input to one dimension before transitioning from the convolutional layer to the fully connected output layer.

Moving on to the generator, this model converts a vector from the latent space that is randomly sampled during training into a 48×48 grayscale synthetic image. For the reasons stated above, we will also use Leaky ReLU as the activation function in this model. We start by fully connected layer in order to interpret the point in the latent space and provide enough activations to reshape it into 128 copies of a low-resolution version of the 24×24 output image. This is then upsampled once, doubling the size and quadrupling the area of the activations using 2-dimensional transpose convolutional layers. Finally, in the output layer, the model employs a tanh activation function.

After defining the discriminator and the generator, we can create the GAN model that combines these two. This larger model aims to train the generator weights based on the discriminator's output and error. The discriminator weights are marked as untrained in this combined model because they are updated separately in the discriminator-only training. The combined model takes the latent points as input and generates synthetic images, which are then fed into the discriminator model, which predicts whether they are from the training set or not.

Each time we update the GAN model, we will need a few real images from the training dataset. For this, we must choose a random selection of images with their associated class label for the discriminator (class = 1), designating that they are real images. The generator model requires inputs that are random variables with a Gaussian distribution. We accomplished this by using a function, which we are going to call "real samples" and given the size of the latent space and the required number of points as parameters, it returns a batch of input samples. To generate new images, we must feed the generator with points from the latent space. To do so, we also create a new function called "fake samples" that generates latent space points and feeds them into the generator model after accepting the generator model and the size of the latent space as inputs. The function returns the created images as well as the discriminator model class label for each non-real image (class=0).

The GAN models can now be fitted. Twenty training epochs are suitable for the model. By setting a batch size of 128 samples, each training epoch includes 101 batches of real and fake samples. One batch of weight updates is created by first updating the discriminator model for a half batch of real samples, followed by a half batch of fake samples. The combined GAN model is then used to update the generator. It is important to mention that the fake samples have the class label set to 1. Consequently, this has the effect of upgrading the generator so that the subsequent batch of "real" samples are produced more successfully and much closer to the actual real ones.

We get three losses from this training. The loss of two discriminators, one trained on real images and one trained on fake samples, and the loss of the generative model. All are reported at the end of each iteration (for each batch) in the training loop. A stable GAN should have both discriminator losses ranging from 0.5 to 0.7 with a maximum of 0.8. Generator loss are usually higher, ranging from 1.0 to 2.0 or more. Both discriminator and generator behavior can start erratically and become more stable as the model approaches equilibrium. With this in mind, we expect the variance of generator and discriminator losses to remain modest, without very sharp fluctuations, otherwise, it's a signal that our model collapsed. Consequently, the generator will most likely generate images that are very similar, even if they come from completely different points in the latent space.

Another major issue that can arise when training the GAN model is convergence failure, which occurs when both losses do not stabilize during the training process, with several peaks of unexpected values in each iteration. This occurs when there is no balance between the discriminator and the generator, and in these cases, the discriminator's loss approaches or reaches null values, where it remains during the training process, whereas the generator's loss is likely to increase progressively, resulting in it continuously producing "junk" images that the discriminator can easily identify as non-real. It is possible for some unstable GANs to show this behavior early on and for some batch updates, and then recover.

3.3.3.DCGAN Improvement 1

Now let's move on to the second phase of the build process, DCGAN Enhancement 1. At this point, some changes are made to both the discriminator and generator models. We decided to follow the stable architecture guidelines for deep convolutional GANs published by Radford et al. Suggested. Al. Article "Unsupervised Representation Learning with Deep Convolutional Generative

Adversarial Networks". In this sense, the first countermeasure for both models was the use of batch normalization. This aims to standardize the activations of the previous layer to zero mean and unit variance which, consequently, stabilizes the training process.

More specifically, normalization is a broad category of methods aimed at making different samples similar to each other which allows machine learning models to learn and generalize well on new data. The most common form of data normalization is to center the data around 0 by subtracting the mean of the data and dividing the data by its standard deviation to give the data a unit standard deviation. This actually assumes that the data follow a normal or gaussian distribution. In our baseline model, we normalized the data before feeding it to the model. But normalizing the data should be a problem after each transformation performed by the network. Even if the data going into a Dense or Conv2D network has a mean of 0 and a variance of 1, there is no reason to believe that all of the following output data follow the same pattern. Batch normalization is a type of layer introduced by Ioffe and Szegedy in 2015, where the main effect is to aid gradient propagation, much like residual connections, thus allowing deeper networks. With this in mind, batch standard units are recommended for discriminator and generator models, except for generator output and discriminator input.

The second change we will make to both models is to use Adam's version of stochastic gradient descent as the optimizer, with a learning rate of 0.0002 and a beta moment value of 0.5 instead of the standard 0.9, rather than RMSProp as we did previously. Adam can be thought of as a fusion of RMSprop and stochastic gradient descent with momentum. It, like RMSprop, uses square gradients to scale the learning rate; additionally, like the SGD with momentum, this optimizer takes advantage of momentum by using a moving average of the gradients rather than the gradients themselves. Adam computes individual learning rates for different parameters, thus becoming an adaptive learning rate method. Adam increases the training speed, and as the gradients become sparser, his bias correction helps slightly outperform the RMSprop at the end of the optimization.

When designing neural network models, weight initialization is critical. Weights are parameters in neural network nodes that are used to compute a weighted sum of the inputs. Neural network models are fitted using a stochastic gradient descent optimization algorithm, which gradually changes the weights in order to minimize the loss function and make useful predictions. To begin the optimization process, this optimization algorithm requires a starting point in the space of possible weight values, and this starting point can determine whether the algorithm converges.

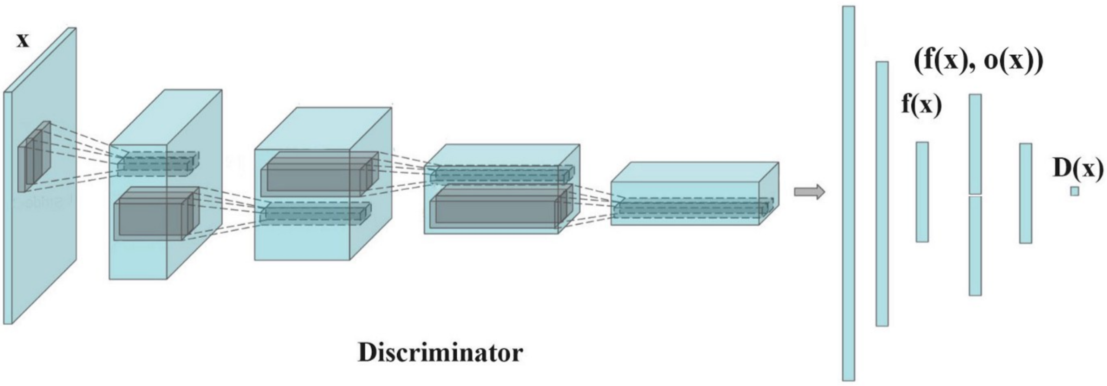
An unstable starting point can cause the algorithm to run in to numerical difficulties and eventually fail. This is where momentum becomes valuable. It addresses issues such as convergence speed and local minima. If there is a local minimum around a certain parameter value and the parameter under consideration is optimized with a small learning rate using SGD, the optimization process will remain at the local minimum rather than reaching the global minimum, increasing the loss. Momentum solves this problem by moving each step based on previous weight updates as well as the current slope (gradient) value.

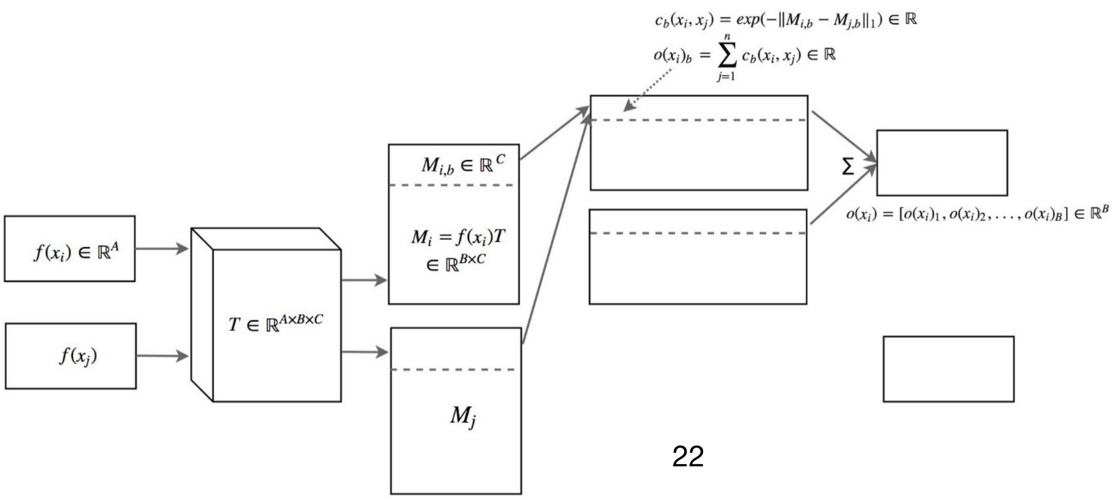
3.3.4.DCGAN Improvement 2

Let us proceed to the third stage of our model's development, "DCGAN Improvement 2," where two strategies will be implemented. Label smoothing, to begin with, is a technique for smoothing classes in a discriminator network. This means that instead of classifying each sample as 1 for real instances or 0 for fake instances, we can have decimal values between 0.7 and 0.9 or 0.1 and 0.3. It is recommended in the case of GANS to use only one-sided-label smoothing with values such as 0.9 in the discriminator for real samples, which is what we are going to do on our model. Smoothing targets can often significantly improve generalization and learning speed because they act as a form of regularization for classification problems by preventing the model from predicting labels too confidently during training, forcing it to learn either a more non-linear function or a linear function with a smaller slope; as a result, extrapolations by the label-smoothed model are less extreme. Overconfidence occurs when a network places nearly all of its probability on a single class in the training set, which is frequently a symptom of overfitting and low entropy because the data is almost entirely concentrated on one value. Label smoothing also reduces mutual information, which tells us how much one random variable tells us about another, resulting in more robustness. It also has an impact on model calibration, which is the process of adjusting model parameters to improve prediction and performance. Finally, label smoothing lowers the likelihood of adversarial examples in GANs. Adversarial attacks are the process of producing adversarial examples that cause a machine learning model to produce unexpected results. In the case of the discriminator, these examples are the synthetic images generated by the generator, which the discriminator wishes to distinguish from images from the training set, which are the real images, keeping in mind that the generator attempts to fool the discriminated by creating increasingly realistic images.

A second strategy used in the model, like smoothed labels, aims to improve the robustness of the training process and reduce the generalization error. When training on small amounts of data, the network tends to memorize all training examples, leading to overfitting as described above. More specifically, this second strategy is to add noise to the labels by injecting some errors into those labels where some fake images are marked as real. Since we were updating the discriminator in the model for real and fake images in separate batches, we chose to randomly flip 5% of the labels in only the fake image batch in each iteration. As a result, random noise is added to the input variables each time a training sample is exposed to the model, making it unique each time. So, adding noise to input samples is a simple form of data augmentation. Since the training patterns are constantly changing, the network cannot remember them, resulting in lower network weights. By limiting the amount of information contained in the input features, you force the network to learn compact representations of the input features and improve accuracy. This technique allows the algorithm to explore its surroundings more effectively, resulting in higher scores and more elegant behavior.

3.3.5.DCGAN Improvement 3





Minibatch Discrimination

Sources from: Medium. 2022. *GAN — Ways to improve GAN performance*. [online] Available at: <https://towardsdatascience.com/gan-ways-to-improve-gan-performance-acf37f9f59b> [Accessed 20 September 2022].

The generator may collapse to a parameter setting in which it continuously outputs the same point, which will cause all the images that are created to have the same appearance. This is one of the most typical failure situations for GAN. When a single mode is on the verge of collapsing, the discriminator's gradient may point in the same direction for numerous points that are quite comparable. There is no coordination between the discriminator's gradients and consequently, there is no mechanism that tells the generator's outputs to become more dissimilar to one another. This is due to the fact that the discriminator performs an individual analysis on each sample. Instead, all of the discriminator's outputs converge on a single point that it now believes to be very realistic. Following the occurrence of collapse, the discriminator understands that this single point originates from the generator, yet gradient descent is unable to differentiate between outputs that are similar. The gradients of the discriminator then drive the single generated point around in space indefinitely, and the mechanism is unable to converge to a distribution with the appropriate amount of entropy. Making it possible for the discriminator to examine many data instances in conjunction with one another and carrying out what is known as "minibatch discrimination" is an easy way to avoid this kind of failure. Let's look at this in a more mathematical perspective:

$$M_i = f(x_i)T$$

The transformation matrix T is used to determine the degree of similarity, denoted by the symbol o(xi), between the image xi and the other images that are part of the same batch.

$$c_b(x_i, x_j) = exp(-\|M_{i,b} - M_{j,b}\|_1) \in \mathbb{R}$$

xi represents the input image , and xj represents the remaining images that are part of the same batch. When we want to transform the features xi into the matrix Mi, which is a BxC matrix, we utilize a transformation matrix called T.

Using the L1-norm and the following equation, we are able to determine the degree of similarity, denoted by the letter c, that exists between images i and j. The L1-norm, commonly know as the Manhattan Distance, is the sum of the magnitudes of all of the vectors that are present within a particular space. It is the simplest approach for measuring the distance between two vectors in the sense that it determines the total amount by which the components of the vectors being compared differ from one another on an absolute scale; more specifically, it is equal to the sum of the lengths of the line segments that are projected onto the coordinate axes from the points that are in between those lines. In accordance with this norm, the same amount of weight is assigned to each of the components that make up the vector.

The similarity between image xi and the other pictures in the batch is denoted by the symbol o(xi).

$$o(x_i)_b = \sum_{j=1}^{n} c_b(x_i, x_j) \in \mathbb{R}$$
$$o(x_i) = [o(x_i)_1, o(x_i)_2, \ldots, o(x_i)_B] \in \mathbb{R}^B$$

These mini batch features are calculated independently for generator and training data samples. As all the previous models that we have created up to this point, the discriminator must produce a single number for each sample that indicates the likelihood that it was derived from the training data. In this way, the discriminator's job is still to determine if a single sample is real or fake data, but it can make use of the other instances in the mini batch as supplementary data.

3.3.6.Conditional DCGAN

Finally, let's hand over "Conditional DCGAN". As mentioned earlier, Conditional GANs are an extension of generative adversarial networks, where generators and discriminators are adjusted during training using some additional information (in this case, sentiment labels). To create this final model, we need to make some changes to all the steps we developed earlier, primarily on discriminator and generator structures. Starting with the discriminant model, a new second input is defined. This takes an integer for the class label of the image. This makes the input image dependent on the given class designation. The class labels are then passed through a 50-element embedding layer. This means that the discriminator model maps each of the dataset's eight classes to a different 50-element vector representation. Using linear activations, the embedding output is propagated to fully connected layers. Activations are converted to a single 48x48 activation map and concatenated to the input image. This gives the next convolutional layer the appearance of a two-channel input image. The generator model must then be updated to accept the class designation. This results in the creation of a point in the latent space that is dependent on the class label provided. Class specifications, like before, are mapped to unique 50-element vectors through embedding layers. It then passes through the fully connected layers using linear activation before being resized. In this case, the fully connected layer activations are scaled to a single 6x6 feature map. A new 6x6 feature map is added to the existing 128, resulting in 129 feature maps, upsampled as in the previous unconditional models. Next, we need to update the composite GAN model. This one takes points in the latent space as inputs and class labels and produces predictions of whether the inputs are real or synthetic. It is important to explicitly connect both the generator's image output and class label input as inputs to the discriminator model. This allows the same class tag input to flow into the generator and into the discriminator. The hard part of converting unconditional GANs to a conditional is done. Finally, we need to update the training process to use the class labels as well. We need to load the dataset and update the real_samples() and real_samples() functions for selecting batches of samples, respectively, to use the actual class labels from the training dataset. The real_samples() function now returns the class label for the image, emotion label, and identifier, class=1.

3.3.7.Results and Evaluation

Note: results from Conditional DCGAN in Appendix 7

We can see the losses leveling off after epochs 4 and 6. The discriminator loss for real images remains between 0.4 and 0.6, whereas it is between 0.2 and 0.4 for fake images and between 0.9 and 2.0 for the generator. The training begins relatively unstable for all models, but it stabilizes over time with values that are less distant from each other for each batch. Despite being more unstable, our base model appears to work as a good initial line that, after being refined in each phase until reaching the end outcome, resulted in a more interesting Conditional GAN.

The generated images improve significantly with each upgrade, and the model does not appear to collapse in the sense that the generated instances do not appear to have a high degree of similarity as if they originated from the same input point into the latent space. The discriminator's loss for false and real images does not approach or reach zero in any iteration, and the generator's loss does not increase progressively over time taking unexpected values; this type of loss is most commonly caused by the generator, which generates faint images that the discriminator can easily identify. We could train for additional epochs for improved image quality, but we couldn't owing to limited computing power. The loss of the generator typically goes up concurrently with an improvement in the picture's overall quality. In order to verify that we are making progress, we turn to manually reviewing the photographs that were generated. This makes it more difficult to compare different models, making it more difficult to choose the model that performs the best in a single run. Additionally, it makes it more difficult to tune. On the other hand, the collapse of the mode is not altogether bad news. When doing a GAN style transfer, our primary goal is to make at least one of the images we work on look better. Indeed, specializing in partial mode collapse can lead to the production of photographs of superior quality.

The generator appears to produce more distinct images from the 4/6 epochs, requiring the discriminator to work harder to identify the generated images from those from the training data. However, we can observe that the images generated by our models are still of moderate quality, not being entirely detectable, indicating that there is still work to be done here so that the generator and discriminator reach a balance point and avoid a situation of convergence failure. However, as the model improves, the quality of the generated photos improves. The accuracy is high if the generated images are similar to genuine photographs on average but different from one another. With a high recall, the generator can create any sample from the training dataset. Harmonic precision and recall are expected. Training stability can degenerate into periods of large variance loss and corresponding low quality generated images; we can see this in our training, but it's very normal as long as our losses do not reach really extreme levels, which is not the case in our situation.

Chapter 4

4. Conclusion

Children who suffer from autism spectrum disorder, have irregularities in the way of communicating, expressing emotions, and interpreting them in others. Because of this, in this project, we proposed the use of generative adversarial networks as a sort of assistive technology to help young children learn about emotions and enhance their relationships and interactions with others. To that end, we showcase an early version of Gan that can generate synthetic images for 8 fundamental emotions, albeit it still has a long way to go before it's fully functional. In order to reach our objective of enhancing this child's quality of life, this serves as a foundation upon which a more complete tool may be built.

Despite being a potentially useful category of generative models, generative adversarial networks have been hampered by unstable training thus far. To address these issues, this paper offers partial solutions: the use of input values in [-1,1] range to make gradients work more effectively with homogeneous and constrained data; the use of Leaky ReLUs for both discriminator and generator to solve the "Dead ReLU" problem; the use of stride convolutions instead of pooling layers; the use o batch normalization to standardize the activations; the use of Adam Optimization to increase training speed, and get sparser gradients; label smoothing and noisy labels to avoid overconfidence discriminator and improve robustness of the training process; mini batch discrimination for to avoid model collapse; and lastly, conditioning Gan with emotion labels to get to our final aim of making a model where autistic kids may pick and choose which emotions they want to observe, investigate, and make on their own.

With this approach, we were able to achieve a stable Conditional GAN that does not collapse, producing images that are distinct from one another and not originated in a single point. We were also able to achieve a model that found a good balance between the generator and discriminator, leaving the latter with the difficult task of distinguishing real from fake images and the first with the task of producing more and more realistic images.

Chapter 5

5. Limitations and further improvements

While the outcomes obtained throughout the project were generally favorable, the whole project would undoubtedly benefit from future enhancements. Due to a lack of processing capacity and accessible data, such as higher-quality training data or additional information, such as Action Units, which could condition our model by various domains, the pictures produced by our proposed GAN are shown at a lower resolution. Despite this, we know that our system is stable; it does not collapse and does not fail to converge.

Exploring more of the latent space in order to accomplish Image-to-Image Translation might potentially be an intriguing addition to this research in the future. We believe that in the long term, we will be able to investigate more sophisticated approaches for stabilizing GANS or perhaps create Progressive Growing Conditional GANS with high-quality synthetic images.

Lastly, we feel that our project served as a guideline toward our ultimate aim. This project will contribute to existing research in the field, or it may be developed in the future for a final outcome that will be able to provide value to a genuine existing issue affecting autistic children.

References

1. Autism Data, 2022. [online] Available at: <https://www.cdc.gov/ncbddd/autism/ data.html>

2. Fischbach, G., 2022. Leo Kanner's 1943 paper on autism. [online] Spectrum | Autism Research News. Available at: <https://www.spectrumnews.org/opinion/ viewpoint/leo-kanners-1943-paper-on-autism/

3. National Institute of Mental Health (NIMH). 2022. Autism Spectrum Disorder. [online] Available at: <https://www.nimh.nih.gov/health/topics/autism-spectrum-disorders-asd>

4. Mazefsky, C., Herrington, J., Siegel, M., Scarpa, A., Maddox, B., Scahill, L. and White, S., 2022. The Role of Emotion Regulation in Autism Spectrum Disorder.

5. Autism Explained, 2022. [online] Available at: <https://psychiatry.org/ psychiatrists/ practice/dsm>

6. A teaching procedure to help children with Autism Spectrum Disorder to label emotion, 2022. [online] Available at: <https://www.researchgate.net/ publication287216676_A_teaching_procedure_to_help_children_with_Autism_Spectrum_Disorder_to_label_emotions>

7. NIDCD. 2022. Autism Spectrum Disorder: Communication Problems in Children. [online] Available at: <https://www.nidcd.nih.gov/health/autism- spectrum-disorder-communication-problems-children> [Accessed 20 September 2022].

8. Capps, L., Kasari, C., Yirmiya, N., & Sigman, M. (1993). Parental perception of emotional expressiveness in children with autism. Journal of Consulting and Clinical Psychology, 61(3), 475.10.1037/0022-006X.61.3.475Search in Google Scholar

9. Baron-Cohen, S., & Wheelwright, S. (2004). The empathy quotient: an investigation of adults with Asperger syndrome or high functioning autism, and normal sex differences.Journal of autism and developmental disorders,34(2), 163-175

10. Knight, V., McKissick, B. R., & Saunders, A. (2013). A review oftechnology- based interventions to teach academic skills to students with autism spectrum disorder. Journal of autism and developmental disorders,43 (11), 2628-2648.

11. Who.int. 2022. Assistive technology. [online] Available at: <https://www.who.int/ news-room/fact-sheets/detail/assistive-technology>

12. Gates, K. (2004). The past perfect promise of facial recognition technology. ACDIS Occasional Paper

13. ACDIS Occasional Paper Zhao, W., Chellappa, R., Phillips, P. J., & Rosenfeld, A. (2003). Face recognition: A literature survey. ACM computing surveys (CSUR),354), 399-458. Zhao, W., Chellappa, R., Phillips, P. J., & Rosenfeld, A. (2003). Face recognition: A literature survey. ACM computing surveys (CSUR),35

14. Baron-Cohen, S., Golan, O., & Ashwin, E. (2009). Can emotion recognition be taught to children with autism spectrum conditions?. Philosophical Transactions of the Royal Society B: Biological Sciences,364 (1535), 3567-3574.

15. Goldsmith, T. R., & LeBlanc, L. A. (2004). Use of technology inInterventions for children with autism. Journal of Early and Inten-sive Behavior Intervention,1(2), 166.

16. Dautenhahn, K., & Werry, I. (2004). Towards interactive robots inautism therapy: Background, motivation and challenges.Pragmatics & Cognition,12(1), 1-35.

17. 18. Reed S, Akata Z, Yan X, Logeswaran L, Schiele B, Lee H: Generative adversarial text to image synthesis. In: ICML (2016)

18. 19. Zhu S, Fidler S, Urtasun R, Lin D, Loy CC: Be your own prada: Fashion synthesis with structural coherence. In: ICCV (2017)

19. Odena A, Olah C, Shlens J: Conditional image synthesis with auxiliary classifier GANs. In: ICML (2017)

20. Ledig C, Theis L, Huszar F, Caballero J, Cunningham A, Acosta A, Aitken A, Tejani A, Totz J, Wang Z, et al.: Photo-realistic single image superresolution using a generative adversarial network. In: CVPR (2017)

21. Mathieu M, Couprie C, LeCun Y: Deep multi-scale video prediction beyond mean square error. In: ICLR (2016)

22. Pathak D, Krahenbuhl P, Donahue J, Darrell T, Efros AA: Context encoders: Feature learning by inpainting. In: CVPR (2016)

23. Isola P, Zhu JY, Zhou T, Efros AA: Image-to-image translation with conditional adversarial networks. In: CVPR (2017)

24. Choi Y, Choi M, Kim M, Ha JW, Kim S, Choo J: Stargan: Unified generative adversarial networks for multi-domain image-to-image
translation. CVPR (2018)

25. Radford, A., Metz, L. and Chintala, S., 2022. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. [online] arXiv.org. Available at: <https://arxiv.org/abs/1511.06434>

26. ExprGAN Facial Expression Editing With Controllable Expression Intensity, 2022. [online] Available at: <https://www.researchgate.net/publication/319662454_ExprGAN_Facial_Expression_Editing_With_Controllable_Expression_Intensity>

27. Pumarola, A., Agudo, A., Martinez, A., Sanfeliu, A. and Moreno-Noguer, F., 2022. GANimation: Anatomically-aware Facial Animation from a Single Image. [online] arXiv.org. Available at: <https://arxiv.org/abs/1807.09251>

28. Tripathy, S., Kannala, J. and Rahtu, E., 2022. ICface: Interpretable and Controllable Face Reenactment Using GANs. [online] arXiv.org. Available at: <https://arxiv.org/abs/1904.01909>

29. F. Qiao, N. Yao, Z. Jiao, Z. Li, H. Chen, and H. Wang, "Geometry-contrastive gan for facial expression transfer," arXiv preprint arXiv:1802.01822, 2018.

30. J. Kossaifi, L. Tran, Y. Panagakis, and M. Pantic, "Gagan: Geometryaware generative adversarial networks," in Proc. CVPR, pp. 878–887, 2018.

31. Ieeexplore.ieee.org. 2022. The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression. [online] Available at: <https://ieeexplore.ieee.org/document/5543262>

32. Ekman, P., 1992. An Argument for Basic Emotions. [online] Paulekman.com. Available at: <https://www.paulekman.com/wp-content/ uploads/2013/07/An-Argument-For-Basic-Emotions.pdf>

33. Darwin-online.org.uk. 2022. [online] Available at: <http://darwin- online.org.uk/content/frameset? pageseq=1&itemID=F1142&viewtype=text>

34. Ekman, P., 2022. Strong evidence for universals in facial expressions: A reply to Russell's mistaken critique.

35. Russell, J., 1997. The Psychology of Facial Expression.

36. Russell, James & Bachorowski, Jo-Anne & Fernandez-Dols, José- Miguel. (2003). Facial and Vocal Expressions of Emotion. Annual Review of Psychology. 54. 329-349. 10.1146/annurev.psych.54.101601.145102.

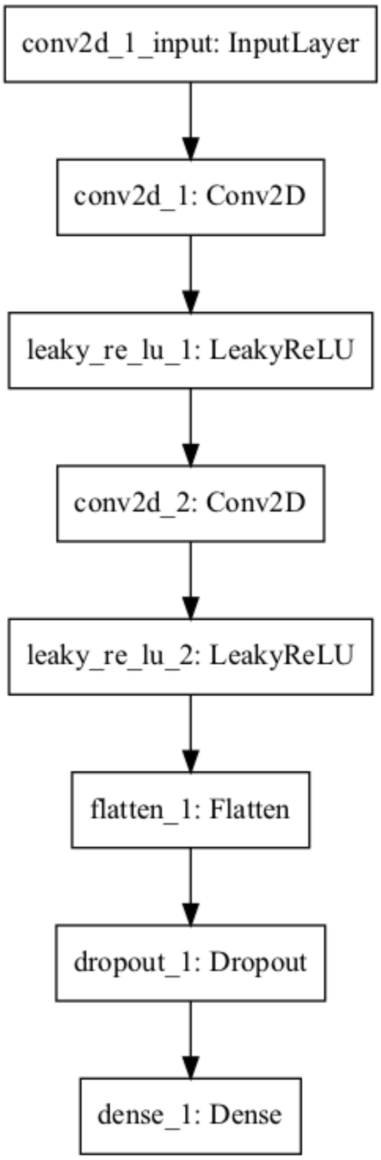
37. Nelson, C. A. (1987). The recognition of facial expressions in the first two years of life: Mechanisms of development. Child Development, 58(4), 889–909. https://doi.org/10.2307/1130530

APPENDIX 2: Generator Baseline Model

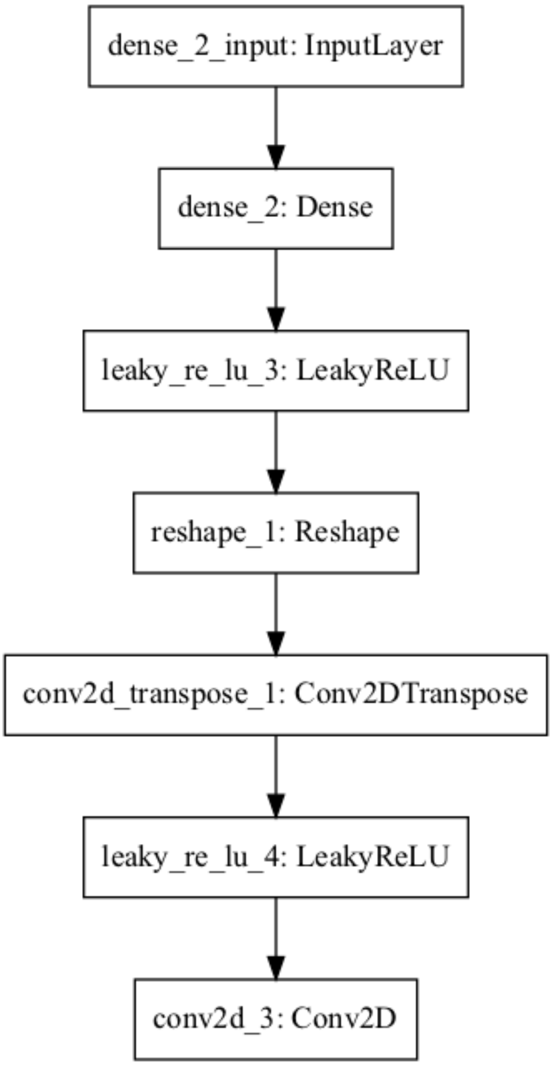
38. Batty, M., & Taylor, M. J. (2006). The development of emotional face processing during childhood. *Developmental Science, 9*(2), 207– 220. https://doi.org/10.1111/j.1467-7687.2006.00480.x

LIST OF APPENDIX

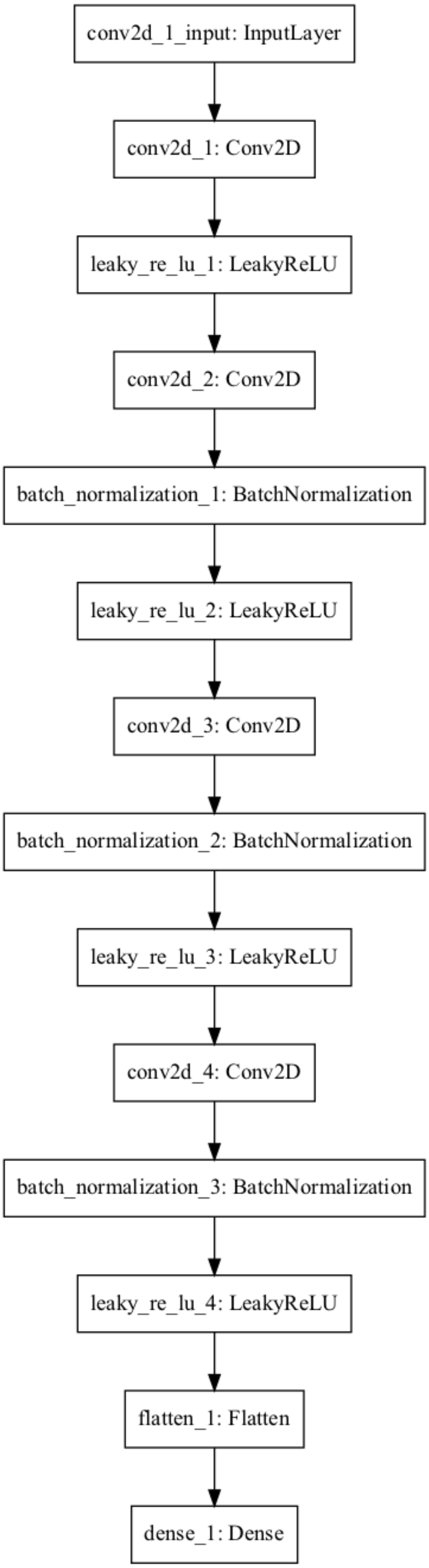
Appendix 1: Discriminator Baseline Model.



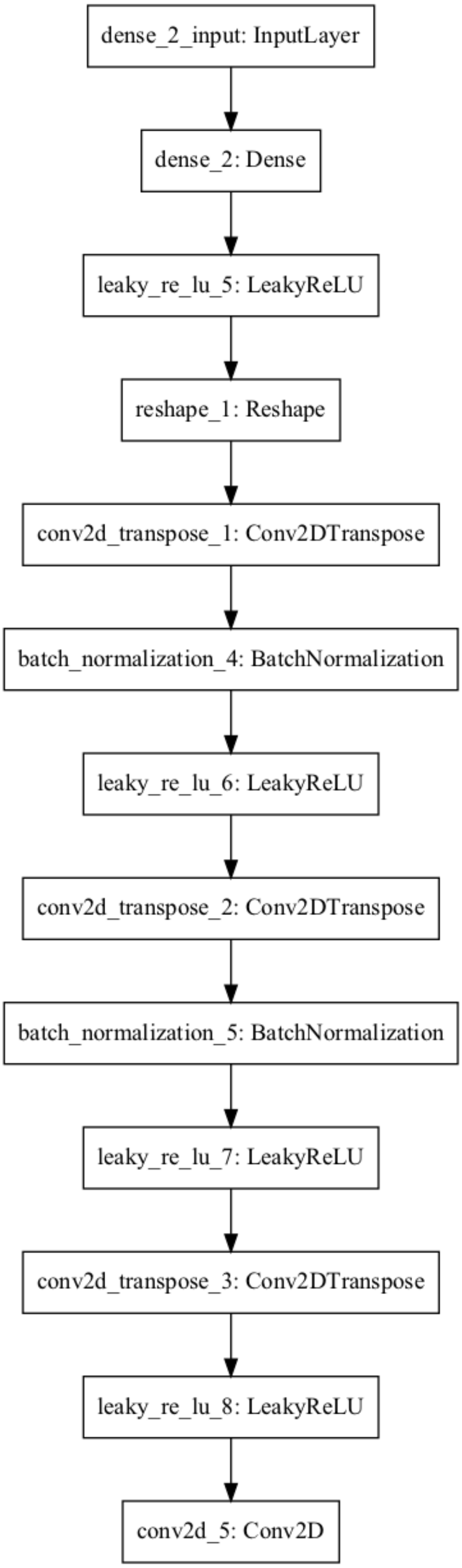
Appendix 2: Generator Baseline Model



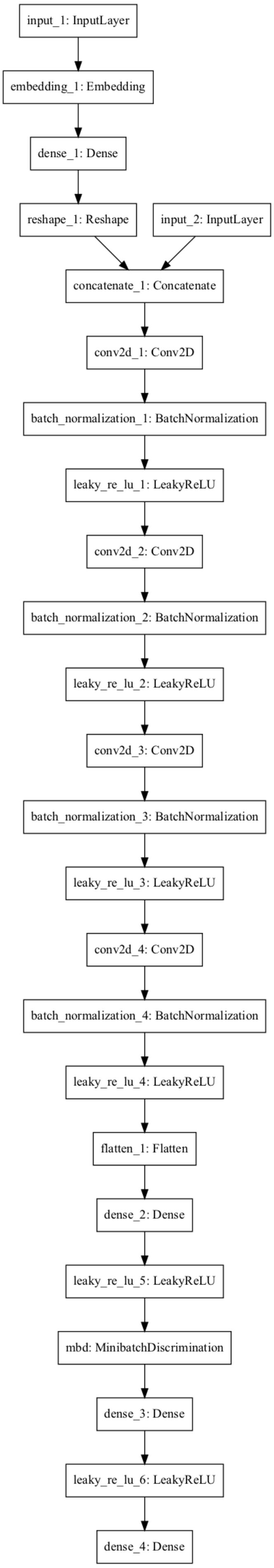
Appendix 3: Discriminator improv.1.



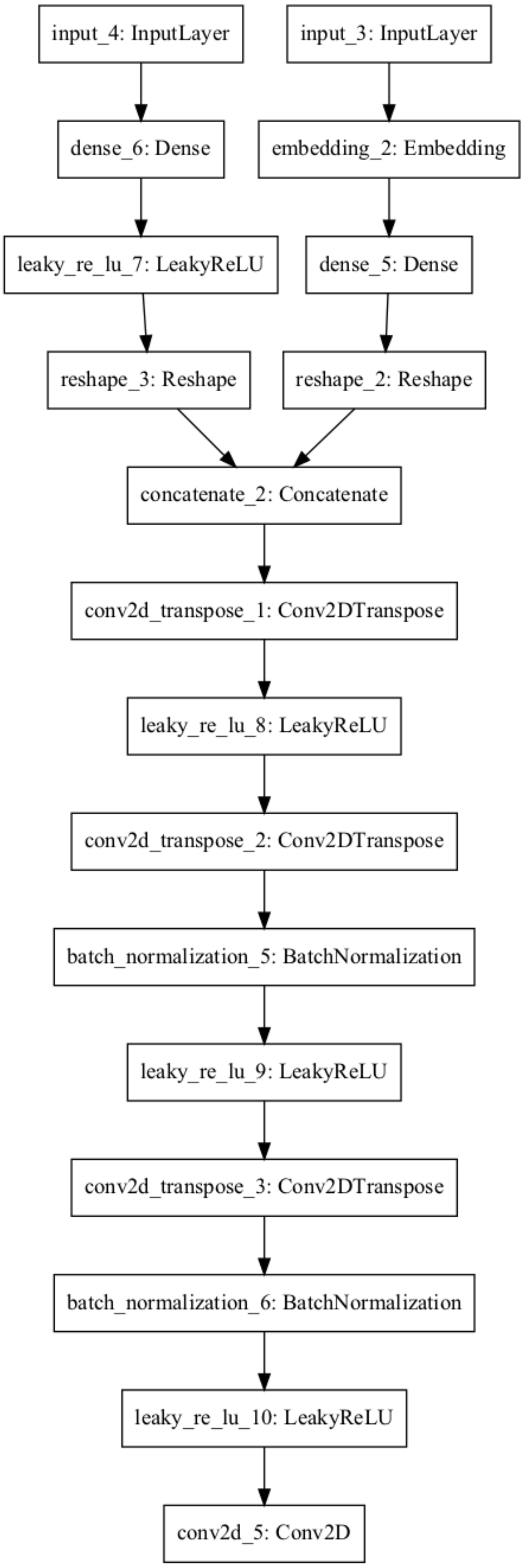
Appendix 4: Generator improv.1



Appendix 5: Conditional Discriminator



Appendix 6: Conditional Generator



Appendix 7: Conditional DCGAN Results per batch

>1, 1/101, d1=2.401, d2=17.812 g=0.000
>1, 2/101, d1=1.107, d2=0.936 g=0.000
>1, 3/101, d1=0.772, d2=0.388 g=0.000
>1, 4/101, d1=0.763, d2=0.201 g=0.000
>1, 5/101, d1=0.692, d2=0.451 g=0.000
>1, 6/101, d1=0.567, d2=0.322 g=0.000
>1, 7/101, d1=0.619, d2=0.368 g=0.005
>1, 8/101, d1=0.515, d2=0.319 g=0.000
>1, 9/101, d1=0.544, d2=0.153 g=0.001
>1, 10/101, d1=0.578, d2=0.425 g=0.000
>1, 11/101, d1=0.657, d2=0.250 g=0.002
>1, 12/101, d1=0.533, d2=0.334 g=0.016
>1, 13/101, d1=0.576, d2=0.349 g=0.000
>1, 14/101, d1=0.587, d2=0.409 g=0.001
>1, 15/101, d1=0.436, d2=0.437 g=0.188
>1, 16/101, d1=0.528, d2=0.333 g=0.033
>1, 17/101, d1=0.488, d2=0.216 g=0.065
>1, 18/101, d1=0.509, d2=0.470 g=0.150
>1, 19/101, d1=0.444, d2=0.431 g=0.317
>1, 20/101, d1=0.499, d2=0.305 g=0.489
>1, 21/101, d1=0.509, d2=0.355 g=0.391
>1, 22/101, d1=0.524, d2=0.449 g=0.320
>1, 23/101, d1=0.436, d2=0.473 g=1.072
>1, 24/101, d1=0.588, d2=0.492 g=0.690
>1, 25/101, d1=0.470, d2=0.521 g=0.655
>1, 26/101, d1=0.474, d2=0.341 g=1.220
>1, 27/101, d1=0.475, d2=0.523 g=1.006
>1, 28/101, d1=0.476, d2=0.245 g=1.252
>1, 29/101, d1=0.443, d2=0.418 g=2.161
>1, 30/101, d1=0.575, d2=0.337 g=2.350
>1, 31/101, d1=0.519, d2=0.474 g=1.541
>1, 32/101, d1=0.470, d2=0.396 g=0.311
>1, 33/101, d1=0.460, d2=0.328 g=1.674
>1, 34/101, d1=0.395, d2=0.288 g=1.573
>1, 35/101, d1=0.420, d2=0.441 g=1.604
>1, 36/101, d1=0.481, d2=0.435 g=1.589
>1, 37/101, d1=0.416, d2=0.421 g=2.488
>1, 38/101, d1=0.422, d2=0.384 g=2.178
>1, 39/101, d1=0.463, d2=0.257 g=1.187
>1, 40/101, d1=0.430, d2=0.425 g=2.090
>1, 41/101, d1=0.591, d2=0.338 g=0.767
>1, 42/101, d1=0.527, d2=0.382 g=2.630
>1, 43/101, d1=0.471, d2=0.320 g=1.315

>1, 44/101, d1=0.433, d2=0.247 g=1.951
>1, 45/101, d1=0.427, d2=0.455 g=2.910
>1, 46/101, d1=0.816, d2=0.400 g=2.299
>1, 47/101, d1=0.481, d2=0.254 g=2.362
>1, 48/101, d1=0.445, d2=0.400 g=2.731
>1, 49/101, d1=0.540, d2=0.366 g=0.799
>1, 50/101, d1=0.483, d2=0.537 g=2.538
>1, 51/101, d1=0.418, d2=0.239 g=3.361
>1, 52/101, d1=0.432, d2=0.279 g=3.457
>1, 53/101, d1=0.462, d2=0.444 g=4.864
>1, 54/101, d1=0.555, d2=0.268 g=4.058
>1, 55/101, d1=0.409, d2=0.199 g=3.481
>1, 56/101, d1=0.408, d2=0.300 g=3.649
>1, 57/101, d1=0.456, d2=0.274 g=3.456
>1, 58/101, d1=0.438, d2=0.352 g=2.825
>1, 59/101, d1=0.383, d2=0.243 g=6.396
>1, 60/101, d1=0.511, d2=0.325 g=3.925
>1, 61/101, d1=0.458, d2=0.356 g=4.433
>1, 62/101, d1=0.423, d2=0.271 g=5.198
>1, 63/101, d1=0.423, d2=0.362 g=4.291
>1, 64/101, d1=0.416, d2=0.286 g=6.302
>1, 65/101, d1=0.495, d2=0.566 g=6.222
>1, 66/101, d1=0.476, d2=0.199 g=5.015
>1, 67/101, d1=0.426, d2=0.211 g=6.613
>1, 68/101, d1=0.388, d2=0.367 g=8.102
>1, 69/101, d1=0.479, d2=0.317 g=7.425
>1, 70/101, d1=0.391, d2=0.301 g=6.363
>1, 71/101, d1=0.387, d2=0.253 g=4.559
>1, 72/101, d1=0.413, d2=0.318 g=4.693
>1, 73/101, d1=0.470, d2=0.290 g=4.607
>1, 74/101, d1=0.413, d2=0.326 g=7.947
>1, 75/101, d1=0.431, d2=0.273 g=4.083
>1, 76/101, d1=0.425, d2=0.317 g=6.658
>1, 77/101, d1=0.503, d2=0.265 g=4.739
>1, 78/101, d1=0.427, d2=0.378 g=6.441
>1, 79/101, d1=0.379, d2=0.299 g=8.851
>1, 80/101, d1=0.489, d2=0.214 g=2.518
>1, 81/101, d1=0.449, d2=0.266 g=4.594
>1, 82/101, d1=0.408, d2=0.314 g=7.191
>1, 83/101, d1=0.454, d2=0.265 g=4.314
>1, 84/101, d1=0.395, d2=0.259 g=4.588
>1, 85/101, d1=0.406, d2=0.372 g=7.475
>1, 86/101, d1=0.534, d2=0.216 g=5.913
>1, 87/101, d1=0.404, d2=0.318 g=7.234
>1, 88/101, d1=0.403, d2=0.290 g=7.784

>1, 89/101, d1=0.513, d2=0.299 g=5.042
>1, 90/101, d1=0.386, d2=0.302 g=7.106
>1, 91/101, d1=0.606, d2=0.916 g=8.078
>1, 92/101, d1=0.750, d2=0.279 g=3.058
>1, 93/101, d1=0.413, d2=0.298 g=3.917
>1, 94/101, d1=0.387, d2=0.291 g=4.785
>1, 95/101, d1=0.379, d2=0.257 g=4.409
>1, 96/101, d1=0.568, d2=1.161 g=9.059
>1, 97/101, d1=1.650, d2=0.942 g=4.585
>1, 98/101, d1=0.529, d2=0.344 g=4.861
>1, 99/101, d1=0.679, d2=0.491 g=2.279
>1, 100/101, d1=0.394, d2=0.301 g=2.229
>1, 101/101, d1=0.475, d2=0.389 g=3.231
>2, 1/101, d1=0.633, d2=0.554 g=3.084
>2, 2/101, d1=0.466, d2=0.227 g=3.850
>2, 3/101, d1=0.442, d2=0.370 g=5.369
>2, 4/101, d1=0.855, d2=0.700 g=4.150
>2, 5/101, d1=0.604, d2=0.280 g=2.847
>2, 6/101, d1=0.423, d2=0.224 g=3.223
>2, 7/101, d1=0.516, d2=0.474 g=5.171
>2, 8/101, d1=0.753, d2=0.307 g=2.689
>2, 9/101, d1=0.466, d2=0.294 g=4.358
>2, 10/101, d1=0.473, d2=0.170 g=2.881
>2, 11/101, d1=0.404, d2=0.174 g=4.071
>2, 12/101, d1=0.502, d2=0.447 g=5.626
>2, 13/101, d1=0.508, d2=0.257 g=2.753
>2, 14/101, d1=0.439, d2=0.412 g=4.809
>2, 15/101, d1=0.585, d2=0.346 g=1.959
>2, 16/101, d1=0.435, d2=0.372 g=5.309
>2, 17/101, d1=1.007, d2=0.628 g=5.305
>2, 18/101, d1=0.713, d2=0.272 g=1.792
>2, 19/101, d1=0.455, d2=0.252 g=2.674
>2, 20/101, d1=0.478, d2=0.317 g=5.212
>2, 21/101, d1=0.679, d2=0.326 g=2.515
>2, 22/101, d1=0.435, d2=0.283 g=5.325
>2, 23/101, d1=0.487, d2=0.373 g=5.315
>2, 24/101, d1=0.596, d2=0.527 g=4.328
>2, 25/101, d1=0.429, d2=0.252 g=3.950
>2, 26/101, d1=0.413, d2=0.295 g=4.070
>2, 27/101, d1=0.474, d2=0.258 g=3.400
>2, 28/101, d1=0.412, d2=0.292 g=4.948
>2, 29/101, d1=0.662, d2=0.461 g=5.215
>2, 30/101, d1=0.444, d2=0.195 g=4.061
>2, 31/101, d1=0.495, d2=0.372 g=3.621
>2, 32/101, d1=0.451, d2=0.278 g=3.527

>2, 33/101, d1=0.481, d2=0.366 g=5.573
>2, 34/101, d1=0.435, d2=0.227 g=4.573
>2, 35/101, d1=0.489, d2=0.558 g=7.643
>2, 36/101, d1=0.483, d2=0.332 g=5.667
>2, 37/101, d1=0.421, d2=0.214 g=5.233
>2, 38/101, d1=0.450, d2=0.342 g=6.151
>2, 39/101, d1=0.410, d2=0.230 g=7.195
>2, 40/101, d1=0.516, d2=0.294 g=4.011
>2, 41/101, d1=0.425, d2=0.237 g=3.929
>2, 42/101, d1=0.419, d2=0.232 g=3.765
>2, 43/101, d1=0.407, d2=0.299 g=3.931
>2, 44/101, d1=0.456, d2=0.309 g=3.861
>2, 45/101, d1=0.442, d2=0.250 g=3.615
>2, 46/101, d1=0.408, d2=0.260 g=5.051
>2, 47/101, d1=0.382, d2=0.179 g=4.452
>2, 48/101, d1=0.455, d2=0.220 g=3.205
>2, 49/101, d1=0.389, d2=0.227 g=4.388
>2, 50/101, d1=0.406, d2=0.237 g=3.304
>2, 51/101, d1=0.373, d2=0.265 g=3.926
>2, 52/101, d1=0.382, d2=0.244 g=3.294
>2, 53/101, d1=0.383, d2=0.221 g=3.868
>2, 54/101, d1=0.406, d2=0.272 g=3.045
>2, 55/101, d1=0.437, d2=0.383 g=4.475
>2, 56/101, d1=0.427, d2=0.241 g=5.135
>2, 57/101, d1=0.372, d2=0.267 g=4.656
>2, 58/101, d1=0.484, d2=0.500 g=6.814
>2, 59/101, d1=0.439, d2=0.255 g=5.422
>2, 60/101, d1=0.382, d2=0.230 g=1.955
>2, 61/101, d1=0.411, d2=0.286 g=3.525
>2, 62/101, d1=0.398, d2=0.209 g=3.266
>2, 63/101, d1=0.418, d2=0.289 g=5.984
>2, 64/101, d1=0.446, d2=0.335 g=4.570
>2, 65/101, d1=0.396, d2=0.223 g=5.120
>2, 66/101, d1=0.392, d2=0.199 g=3.504
>2, 67/101, d1=0.413, d2=0.335 g=3.396
>2, 68/101, d1=0.405, d2=0.298 g=3.964
>2, 69/101, d1=0.436, d2=0.233 g=3.235
>2, 70/101, d1=0.399, d2=0.221 g=4.789
>2, 71/101, d1=0.545, d2=0.308 g=2.733
>2, 72/101, d1=0.388, d2=0.276 g=4.139
>2, 73/101, d1=0.481, d2=0.277 g=3.592
>2, 74/101, d1=0.408, d2=0.275 g=3.112
>2, 75/101, d1=0.423, d2=0.441 g=6.295
>2, 76/101, d1=0.837, d2=0.842 g=7.634
>2, 77/101, d1=0.590, d2=0.333 g=5.235

>2, 78/101, d1=0.410, d2=0.219 g=2.142
>2, 79/101, d1=0.408, d2=0.472 g=6.171
>2, 80/101, d1=0.903, d2=0.689 g=9.245
>2, 81/101, d1=0.615, d2=0.237 g=4.774
>2, 82/101, d1=0.430, d2=0.185 g=4.341
>2, 83/101, d1=0.437, d2=0.239 g=4.227
>2, 84/101, d1=0.408, d2=0.216 g=3.412
>2, 85/101, d1=0.391, d2=0.167 g=2.793
>2, 86/101, d1=0.393, d2=0.305 g=5.094
>2, 87/101, d1=0.464, d2=0.243 g=3.145
>2, 88/101, d1=0.410, d2=0.274 g=2.755
>2, 89/101, d1=0.427, d2=0.227 g=3.500
>2, 90/101, d1=0.377, d2=0.193 g=3.757
>2, 91/101, d1=0.644, d2=0.375 g=3.462
>2, 92/101, d1=0.468, d2=0.294 g=6.157
>2, 93/101, d1=0.393, d2=0.275 g=2.227
>2, 94/101, d1=0.399, d2=0.240 g=1.636
>2, 95/101, d1=0.373, d2=0.349 g=2.954
>2, 96/101, d1=0.468, d2=0.346 g=4.225
>2, 97/101, d1=0.355, d2=0.239 g=4.865
>2, 98/101, d1=0.573, d2=0.471 g=6.669
>2, 99/101, d1=0.392, d2=0.291 g=5.695
>2, 100/101, d1=0.376, d2=0.257 g=2.252
>2, 101/101, d1=0.376, d2=0.328 g=2.626
>3, 1/101, d1=0.400, d2=0.233 g=3.570
>3, 2/101, d1=0.410, d2=0.310 g=2.969
>3, 3/101, d1=0.376, d2=0.258 g=2.457
>3, 4/101, d1=0.396, d2=0.242 g=3.230
>3, 5/101, d1=0.364, d2=0.228 g=3.953
>3, 6/101, d1=0.402, d2=0.222 g=2.704
>3, 7/101, d1=0.370, d2=0.244 g=3.332
>3, 8/101, d1=0.371, d2=0.227 g=4.475
>3, 9/101, d1=0.430, d2=0.241 g=2.880
>3, 10/101, d1=0.376, d2=0.257 g=3.407

>3, 11/101, d1=0.391, d2=0.272 g=3.747
>3, 12/101, d1=0.382, d2=0.225 g=3.599
>3, 13/101, d1=0.373, d2=0.238 g=2.060
>3, 14/101, d1=0.384, d2=0.164 g=2.564
>3, 15/101, d1=0.361, d2=0.228 g=4.277
>3, 16/101, d1=0.436, d2=0.298 g=3.470
>3, 17/101, d1=0.369, d2=0.300 g=3.914
>3, 18/101, d1=0.385, d2=0.233 g=3.949
>3, 19/101, d1=0.355, d2=0.234 g=3.955
>3, 20/101, d1=0.383, d2=0.257 g=6.661

>3, 21/101, d1=0.373, d2=0.240 g=4.810
>3, 22/101, d1=0.380, d2=0.218 g=3.827
>3, 23/101, d1=0.357, d2=0.312 g=3.353
>3, 24/101, d1=0.381, d2=0.315 g=3.040
>3, 25/101, d1=0.371, d2=0.201 g=4.068
>3, 26/101, d1=0.406, d2=0.221 g=2.524
>3, 27/101, d1=0.378, d2=0.239 g=1.845
>3, 28/101, d1=0.378, d2=0.361 g=4.884
>3, 29/101, d1=0.359, d2=0.229 g=6.102
>3, 30/101, d1=0.392, d2=0.267 g=1.839
>3, 31/101, d1=0.371, d2=0.211 g=2.071
>3, 32/101, d1=0.377, d2=0.214 g=3.512
>3, 33/101, d1=0.359, d2=0.255 g=2.208
>3, 34/101, d1=0.374, d2=0.211 g=2.790
>3, 35/101, d1=0.380, d2=0.231 g=1.657
>3, 36/101, d1=0.385, d2=0.237 g=3.333
>3, 37/101, d1=0.360, d2=0.267 g=4.681
>3, 38/101, d1=0.422, d2=0.286 g=1.359
>3, 39/101, d1=0.410, d2=0.356 g=4.887
>3, 40/101, d1=0.393, d2=0.279 g=3.952
>3, 41/101, d1=0.417, d2=0.254 g=0.747
>3, 42/101, d1=0.395, d2=0.242 g=4.373
>3, 43/101, d1=0.366, d2=0.228 g=3.923
>3, 44/101, d1=0.445, d2=0.327 g=1.927
>3, 45/101, d1=0.391, d2=0.248 g=5.157
>3, 46/101, d1=0.387, d2=0.238 g=2.873
>3, 47/101, d1=0.385, d2=0.226 g=2.188
>3, 48/101, d1=0.350, d2=0.212 g=2.488
>3, 49/101, d1=0.363, d2=0.244 g=3.018
>3, 50/101, d1=0.372, d2=0.248 g=2.124
>3, 51/101, d1=0.399, d2=0.298 g=4.705
>3, 52/101, d1=0.389, d2=0.269 g=3.009
>3, 53/101, d1=0.368, d2=0.220 g=3.757
>3, 54/101, d1=0.487, d2=0.468 g=9.977
>3, 55/101, d1=1.462, d2=1.692 g=8.882
>3, 56/101, d1=2.193, d2=0.289 g=0.794
>3, 57/101, d1=0.459, d2=0.394 g=2.566
>3, 58/101, d1=0.459, d2=0.214 g=2.467
>3, 59/101, d1=0.394, d2=0.233 g=2.190
>3, 60/101, d1=0.422, d2=0.189 g=1.549
>3, 61/101, d1=0.400, d2=0.245 g=2.452
>3, 62/101, d1=0.465, d2=0.286 g=2.228
>3, 63/101, d1=0.700, d2=0.836 g=4.553
>3, 64/101, d1=0.538, d2=0.248 g=3.326
>3, 65/101, d1=0.488, d2=0.274 g=2.072

>3, 66/101, d1=0.375, d2=0.237 g=3.423
>3, 67/101, d1=0.550, d2=0.225 g=1.308
>3, 68/101, d1=0.396, d2=0.380 g=3.562
>3, 69/101, d1=0.628, d2=0.500 g=4.316
>3, 70/101, d1=0.411, d2=0.252 g=3.476
>3, 71/101, d1=0.394, d2=0.210 g=3.123
>3, 72/101, d1=0.492, d2=0.259 g=3.758
>3, 73/101, d1=0.367, d2=0.246 g=2.704
>3, 74/101, d1=0.406, d2=0.218 g=1.674
>3, 75/101, d1=0.373, d2=0.348 g=1.643
>3, 76/101, d1=0.408, d2=0.310 g=4.592
>3, 77/101, d1=0.515, d2=0.209 g=1.541
>3, 78/101, d1=0.384, d2=0.267 g=3.242
>3, 79/101, d1=0.480, d2=0.451 g=3.975
>3, 80/101, d1=0.514, d2=0.230 g=0.936
>3, 81/101, d1=0.406, d2=0.460 g=9.379
>3, 82/101, d1=0.728, d2=0.215 g=1.797
>3, 83/101, d1=0.422, d2=0.415 g=5.559
>3, 84/101, d1=0.372, d2=0.243 g=4.678
>3, 85/101, d1=0.376, d2=0.189 g=1.884
>3, 86/101, d1=0.399, d2=0.231 g=1.653
>3, 87/101, d1=0.421, d2=0.259 g=2.651
>3, 88/101, d1=0.373, d2=0.189 g=2.249
>3, 89/101, d1=0.424, d2=0.229 g=0.893
>3, 90/101, d1=0.351, d2=0.201 g=2.511
>3, 91/101, d1=0.383, d2=0.228 g=1.531
>3, 92/101, d1=0.365, d2=0.261 g=2.759
>3, 93/101, d1=0.373, d2=0.206 g=3.451
>3, 94/101, d1=0.370, d2=0.208 g=2.189
>3, 95/101, d1=0.395, d2=0.267 g=5.431
>3, 96/101, d1=0.496, d2=0.236 g=2.826
>3, 97/101, d1=0.427, d2=0.181 g=2.687
>3, 98/101, d1=0.367, d2=0.352 g=7.750
>3, 99/101, d1=0.889, d2=0.437 g=5.210
>3, 100/101, d1=0.369, d2=0.265 g=4.666
>3, 101/101, d1=0.377, d2=0.237 g=3.097
>4, 1/101, d1=0.413, d2=0.362 g=3.968
>4, 2/101, d1=0.383, d2=0.229 g=3.117
>4, 3/101, d1=0.376, d2=0.303 g=1.750
>4, 4/101, d1=0.358, d2=0.257 g=2.460
>4, 5/101, d1=0.556, d2=0.508 g=3.766
>4, 6/101, d1=0.384, d2=0.228 g=4.303
>4, 7/101, d1=0.445, d2=0.247 g=1.835
>4, 8/101, d1=0.378, d2=0.225 g=2.286
>4, 9/101, d1=0.380, d2=0.207 g=0.968

>4, 10/101, d1=0.376, d2=0.289 g=0.578
>4, 11/101, d1=0.379, d2=0.251 g=0.380
>4, 12/101, d1=0.450, d2=0.196 g=0.020
>4, 13/101, d1=0.404, d2=0.135 g=0.290
>4, 14/101, d1=0.366, d2=0.175 g=0.071
>4, 15/101, d1=0.365, d2=0.261 g=0.120
>4, 16/101, d1=0.371, d2=0.178 g=1.483
>4, 17/101, d1=0.715, d2=1.103 g=10.674
>4, 18/101, d1=2.073, d2=0.278 g=0.247
>4, 19/101, d1=0.493, d2=0.641 g=1.707
>4, 20/101, d1=0.438, d2=0.212 g=1.518
>4, 21/101, d1=0.407, d2=0.229 g=0.625
>4, 22/101, d1=0.399, d2=0.371 g=1.439
>4, 23/101, d1=0.633, d2=0.337 g=1.274
>4, 24/101, d1=0.423, d2=0.221 g=0.232
>4, 25/101, d1=0.387, d2=0.257 g=0.098
>4, 26/101, d1=0.374, d2=0.197 g=0.124
>4, 27/101, d1=0.365, d2=0.185 g=0.535
>4, 28/101, d1=0.442, d2=0.440 g=5.665
>4, 29/101, d1=0.419, d2=0.249 g=4.422
>4, 30/101, d1=0.398, d2=0.362 g=9.589
>4, 31/101, d1=0.558, d2=0.242 g=5.221
>4, 32/101, d1=0.398, d2=0.238 g=5.115
>4, 33/101, d1=0.371, d2=0.171 g=4.008
>4, 34/101, d1=0.389, d2=0.345 g=4.909
>4, 35/101, d1=0.451, d2=0.273 g=4.566
>4, 36/101, d1=0.369, d2=0.155 g=4.377
>4, 37/101, d1=0.397, d2=0.313 g=6.318
>4, 38/101, d1=0.478, d2=0.255 g=1.659
>4, 39/101, d1=0.407, d2=0.320 g=5.857
>4, 40/101, d1=0.395, d2=0.248 g=3.462
>4, 41/101, d1=0.387, d2=0.208 g=0.919
>4, 42/101, d1=0.367, d2=0.303 g=3.976
>4, 43/101, d1=0.723, d2=1.562 g=13.778
>4, 44/101, d1=2.004, d2=0.313 g=4.372
>4, 45/101, d1=0.415, d2=0.230 g=1.413
>4, 46/101, d1=0.392, d2=0.282 g=1.982
>4, 47/101, d1=0.379, d2=0.239 g=1.890
>4, 48/101, d1=0.402, d2=0.225 g=1.286
>4, 49/101, d1=0.369, d2=0.202 g=1.623
>4, 50/101, d1=0.406, d2=0.224 g=1.374
>4, 51/101, d1=0.426, d2=0.362 g=1.608
>4, 52/101, d1=0.389, d2=0.139 g=1.406
>4, 53/101, d1=0.435, d2=0.301 g=1.239
>4, 54/101, d1=0.439, d2=0.268 g=1.705

>4, 55/101, d1=0.548, d2=0.424 g=1.056
>4, 56/101, d1=0.383, d2=0.138 g=0.966
>4, 57/101, d1=0.462, d2=0.499 g=2.343
>4, 58/101, d1=0.639, d2=0.404 g=1.851
>4, 59/101, d1=0.515, d2=0.310 g=2.102
>4, 60/101, d1=0.452, d2=0.311 g=1.900
>4, 61/101, d1=0.410, d2=0.330 g=2.757
>4, 62/101, d1=0.396, d2=0.244 g=2.477
>4, 63/101, d1=0.471, d2=0.362 g=2.618
>4, 64/101, d1=0.410, d2=0.183 g=2.091
>4, 65/101, d1=0.450, d2=0.301 g=1.241
>4, 66/101, d1=0.367, d2=0.214 g=2.724
>4, 67/101, d1=0.474, d2=0.245 g=2.313
>4, 68/101, d1=0.375, d2=0.223 g=2.280
>4, 69/101, d1=0.455, d2=0.613 g=5.047
>4, 70/101, d1=1.013, d2=0.472 g=2.412
>4, 71/101, d1=0.509, d2=0.333 g=2.462
>4, 72/101, d1=0.388, d2=0.136 g=2.728
>4, 73/101, d1=0.551, d2=0.581 g=3.796
>4, 74/101, d1=0.521, d2=0.207 g=2.204
>4, 75/101, d1=0.414, d2=0.288 g=1.839
>4, 76/101, d1=0.413, d2=0.259 g=1.685
>4, 77/101, d1=0.602, d2=0.957 g=5.811
>4, 78/101, d1=1.269, d2=0.215 g=0.712
>4, 79/101, d1=0.415, d2=0.500 g=2.892
>4, 80/101, d1=0.461, d2=0.184 g=2.204
>4, 81/101, d1=0.439, d2=0.511 g=2.526
>4, 82/101, d1=0.404, d2=0.203 g=1.943
>4, 83/101, d1=0.456, d2=0.269 g=1.393
>4, 84/101, d1=0.422, d2=0.341 g=2.017
>4, 85/101, d1=0.478, d2=0.278 g=1.684
>4, 86/101, d1=0.423, d2=0.361 g=3.478
>4, 87/101, d1=0.709, d2=0.594 g=4.361
>4, 88/101, d1=0.430, d2=0.183 g=4.495
>4, 89/101, d1=0.643, d2=0.393 g=3.897
>4, 90/101, d1=0.428, d2=0.192 g=3.216
>4, 91/101, d1=0.419, d2=0.195 g=1.374
>4, 92/101, d1=0.417, d2=0.383 g=3.223
>4, 93/101, d1=0.548, d2=0.432 g=3.240
>4, 94/101, d1=0.929, d2=0.727 g=3.705
>4, 95/101, d1=0.544, d2=0.164 g=1.941
>4, 96/101, d1=0.447, d2=0.471 g=2.819
>4, 97/101, d1=0.492, d2=0.200 g=2.359
>4, 98/101, d1=0.463, d2=0.417 g=3.027
>4, 99/101, d1=0.593, d2=0.361 g=3.378

>4, 100/101, d1=0.499, d2=0.225 g=2.695
>4, 101/101, d1=0.406, d2=0.373 g=3.854
>5, 1/101, d1=0.557, d2=0.391 g=2.351
>5, 2/101, d1=0.655, d2=0.487 g=0.895
>5, 3/101, d1=0.406, d2=0.191 g=1.155
>5, 4/101, d1=0.548, d2=0.361 g=0.258
>5, 5/101, d1=0.419, d2=0.162 g=0.619
>5, 6/101, d1=0.446, d2=0.246 g=0.206
>5, 7/101, d1=0.378, d2=0.275 g=2.354
>5, 8/101, d1=0.798, d2=0.552 g=2.641
>5, 9/101, d1=0.414, d2=0.204 g=3.563
>5, 10/101, d1=0.583, d2=0.659 g=6.630
>5, 11/101, d1=1.013, d2=0.303 g=1.036
>5, 12/101, d1=0.461, d2=0.335 g=3.388
>5, 13/101, d1=0.377, d2=0.233 g=5.394
>5, 14/101, d1=0.463, d2=0.211 g=1.395
>5, 15/101, d1=0.382, d2=0.260 g=0.953
>5, 16/101, d1=0.372, d2=0.213 g=1.827
>5, 17/101, d1=0.555, d2=0.400 g=1.427
>5, 18/101, d1=0.483, d2=0.380 g=1.833
>5, 19/101, d1=0.421, d2=0.236 g=1.218

>5, 20/101, d1=0.497, d2=0.294 g=0.575
>5, 21/101, d1=0.407, d2=0.292 g=2.740
>5, 22/101, d1=0.559, d2=0.274 g=1.299
>5, 23/101, d1=0.444, d2=0.318 g=1.473
>5, 24/101, d1=0.413, d2=0.248 g=1.466
>5, 25/101, d1=0.397, d2=0.248 g=0.988
>5, 26/101, d1=0.440, d2=0.357 g=1.080
>5, 27/101, d1=0.388, d2=0.264 g=3.307
>5, 28/101, d1=0.734, d2=0.313 g=0.513
>5, 29/101, d1=0.442, d2=0.345 g=3.795
>5, 30/101, d1=0.547, d2=0.167 g=1.619
>5, 31/101, d1=0.429, d2=0.557 g=5.070
>5, 32/101, d1=0.566, d2=0.219 g=3.099
>5, 33/101, d1=0.442, d2=0.265 g=3.706
>5, 34/101, d1=0.416, d2=0.264 g=3.784
>5, 35/101, d1=0.566, d2=0.620 g=6.851
>5, 36/101, d1=0.542, d2=0.186 g=5.144
>5, 37/101, d1=0.444, d2=0.579 g=8.514
>5, 38/101, d1=1.153, d2=0.428 g=3.791
>5, 39/101, d1=0.423, d2=0.219 g=4.340
>5, 40/101, d1=0.421, d2=0.210 g=3.500
>5, 41/101, d1=0.500, d2=0.452 g=3.983
>5, 42/101, d1=0.425, d2=0.240 g=3.602

>5, 43/101, d1=0.514, d2=0.332 g=1.624
>5, 44/101, d1=0.380, d2=0.298 g=4.107
>5, 45/101, d1=0.614, d2=0.326 g=2.432
>5, 46/101, d1=0.458, d2=0.325 g=2.625
>5, 47/101, d1=0.439, d2=0.220 g=1.539
>5, 48/101, d1=0.500, d2=0.505 g=3.772
>5, 49/101, d1=0.454, d2=0.225 g=1.427
>5, 50/101, d1=0.466, d2=0.314 g=0.681
>5, 51/101, d1=0.412, d2=0.228 g=0.499
>5, 52/101, d1=0.420, d2=0.235 g=0.192
>5, 53/101, d1=0.454, d2=0.253 g=0.163
>5, 54/101, d1=0.410, d2=0.172 g=0.792
>5, 55/101, d1=0.471, d2=0.304 g=0.488
>5, 56/101, d1=0.438, d2=0.287 g=0.337
>5, 57/101, d1=0.404, d2=0.299 g=0.715
>5, 58/101, d1=0.422, d2=0.300 g=3.083
>5, 59/101, d1=0.520, d2=0.329 g=3.318
>5, 60/101, d1=0.480, d2=0.312 g=3.637
>5, 61/101, d1=0.443, d2=0.271 g=3.325
>5, 62/101, d1=0.438, d2=0.278 g=2.799
>5, 63/101, d1=0.409, d2=0.287 g=3.398
>5, 64/101, d1=0.377, d2=0.211 g=3.497
>5, 65/101, d1=0.474, d2=0.434 g=5.298
>5, 66/101, d1=0.665, d2=0.217 g=2.168
>5, 67/101, d1=0.451, d2=0.292 g=4.945
>5, 68/101, d1=0.546, d2=0.218 g=3.199
>5, 69/101, d1=0.408, d2=0.240 g=3.184
>5, 70/101, d1=0.453, d2=0.276 g=3.807
>5, 71/101, d1=0.456, d2=0.213 g=2.346
>5, 72/101, d1=0.392, d2=0.237 g=3.328
>5, 73/101, d1=0.432, d2=0.224 g=3.450
>5, 74/101, d1=0.671, d2=0.931 g=5.125
>5, 75/101, d1=0.645, d2=0.227 g=2.076
>5, 76/101, d1=0.408, d2=0.341 g=2.579
>5, 77/101, d1=0.397, d2=0.274 g=2.132
>5, 78/101, d1=0.451, d2=0.348 g=1.909
>5, 79/101, d1=0.446, d2=0.279 g=0.735
>5, 80/101, d1=0.378, d2=0.202 g=0.662
>5, 81/101, d1=0.473, d2=0.262 g=0.581
>5, 82/101, d1=0.384, d2=0.213 g=0.695
>5, 83/101, d1=0.456, d2=0.399 g=1.026
>5, 84/101, d1=0.411, d2=0.189 g=1.775
>5, 85/101, d1=0.416, d2=0.224 g=0.757
>5, 86/101, d1=0.458, d2=0.222 g=1.102
>5, 87/101, d1=0.466, d2=0.265 g=1.165

>5, 88/101, d1=0.422, d2=0.195 g=0.599
>5, 89/101, d1=0.388, d2=0.237 g=1.056
>5, 90/101, d1=0.384, d2=0.228 g=1.253
>5, 91/101, d1=0.443, d2=0.472 g=3.248
>5, 92/101, d1=0.829, d2=0.422 g=2.376
>5, 93/101, d1=0.439, d2=0.289 g=4.163
>5, 94/101, d1=0.451, d2=0.209 g=2.767
>5, 95/101, d1=0.413, d2=0.263 g=3.568
>5, 96/101, d1=0.424, d2=0.209 g=2.077
>5, 97/101, d1=0.380, d2=0.297 g=3.786
>5, 98/101, d1=0.476, d2=0.225 g=3.323
>5, 99/101, d1=0.406, d2=0.234 g=4.572
>5, 100/101, d1=0.494, d2=0.341 g=2.663
>5, 101/101, d1=0.435, d2=0.282 g=4.357
>6, 1/101, d1=0.397, d2=0.242 g=3.766
>6, 2/101, d1=0.494, d2=0.335 g=4.172
>6, 3/101, d1=0.434, d2=0.239 g=3.961
>6, 4/101, d1=0.533, d2=0.511 g=5.167
>6, 5/101, d1=0.637, d2=0.222 g=2.275
>6, 6/101, d1=0.450, d2=0.324 g=4.071
>6, 7/101, d1=0.421, d2=0.153 g=4.547
>6, 8/101, d1=0.733, d2=0.294 g=0.670
>6, 9/101, d1=0.454, d2=0.269 g=1.261
>6, 10/101, d1=0.462, d2=0.238 g=0.556
>6, 11/101, d1=0.424, d2=0.188 g=0.660
>6, 12/101, d1=0.410, d2=0.229 g=0.485
>6, 13/101, d1=0.408, d2=0.232 g=0.304
>6, 14/101, d1=0.384, d2=0.209 g=0.693
>6, 15/101, d1=0.375, d2=0.287 g=0.924
>6, 16/101, d1=0.376, d2=0.362 g=2.577
>6, 17/101, d1=1.021, d2=1.451 g=6.419
>6, 18/101, d1=1.113, d2=0.252 g=2.571
>6, 19/101, d1=0.430, d2=0.226 g=2.048
>6, 20/101, d1=0.405, d2=0.320 g=2.867
>6, 21/101, d1=0.461, d2=0.302 g=2.136
>6, 22/101, d1=0.392, d2=0.247 g=1.966
>6, 23/101, d1=0.430, d2=0.226 g=0.908
>6, 24/101, d1=0.396, d2=0.269 g=1.476
>6, 25/101, d1=0.412, d2=0.317 g=1.124
>6, 26/101, d1=0.405, d2=0.317 g=1.411
>6, 27/101, d1=0.511, d2=0.500 g=3.344
>6, 28/101, d1=0.477, d2=0.188 g=2.403
>6, 29/101, d1=0.515, d2=0.352 g=2.325
>6, 30/101, d1=0.398, d2=0.219 g=2.336
>6, 31/101, d1=0.461, d2=0.264 g=3.144

>6, 32/101, d1=0.417, d2=0.205 g=2.085
>6, 33/101, d1=0.404, d2=0.279 g=1.379
>6, 34/101, d1=0.411, d2=0.288 g=2.720
>6, 35/101, d1=0.420, d2=0.198 g=1.936
>6, 36/101, d1=0.440, d2=0.270 g=1.358
>6, 37/101, d1=0.402, d2=0.223 g=2.803
>6, 38/101, d1=0.452, d2=0.286 g=2.725
>6, 39/101, d1=0.382, d2=0.227 g=2.341
>6, 40/101, d1=0.393, d2=0.266 g=2.656
>6, 41/101, d1=0.394, d2=0.219 g=3.134
>6, 42/101, d1=0.384, d2=0.266 g=2.534
>6, 43/101, d1=0.392, d2=0.291 g=4.317
>6, 44/101, d1=0.383, d2=0.191 g=4.791
>6, 45/101, d1=0.401, d2=0.253 g=2.945
>6, 46/101, d1=0.369, d2=0.205 g=3.150
>6, 47/101, d1=0.404, d2=0.224 g=2.135
>6, 48/101, d1=0.381, d2=0.197 g=3.382
>6, 49/101, d1=0.454, d2=0.181 g=2.300
>6, 50/101, d1=0.380, d2=0.253 g=1.478
>6, 51/101, d1=0.368, d2=0.302 g=2.237
>6, 52/101, d1=0.398, d2=0.214 g=1.579
>6, 53/101, d1=0.371, d2=0.220 g=0.499
>6, 54/101, d1=0.361, d2=0.225 g=0.445
>6, 55/101, d1=0.400, d2=0.249 g=0.341
>6, 56/101, d1=0.381, d2=0.258 g=0.299
>6, 57/101, d1=0.382, d2=0.318 g=0.814
>6, 58/101, d1=0.364, d2=0.222 g=0.342
>6, 59/101, d1=0.356, d2=0.217 g=0.998
>6, 60/101, d1=0.400, d2=0.281 g=0.560
>6, 61/101, d1=0.357, d2=0.192 g=0.916
>6, 62/101, d1=0.401, d2=0.252 g=0.394
>6, 63/101, d1=0.365, d2=0.215 g=0.428
>6, 64/101, d1=0.448, d2=0.587 g=3.113
>6, 65/101, d1=0.652, d2=0.172 g=0.893
>6, 66/101, d1=0.391, d2=0.371 g=2.455
>6, 67/101, d1=0.651, d2=0.269 g=0.933
>6, 68/101, d1=0.376, d2=0.333 g=2.250
>6, 69/101, d1=0.372, d2=0.204 g=2.178
>6, 70/101, d1=0.476, d2=0.322 g=1.621
>6, 71/101, d1=0.481, d2=0.416 g=2.429
>6, 72/101, d1=0.560, d2=0.251 g=1.890
>6, 73/101, d1=0.394, d2=0.290 g=3.675
>6, 74/101, d1=0.458, d2=0.235 g=2.870
>6, 75/101, d1=0.424, d2=0.366 g=5.536
>6, 76/101, d1=0.460, d2=0.233 g=3.425

>6, 77/101, d1=0.382, d2=0.274 g=4.138
>6, 78/101, d1=0.489, d2=0.294 g=2.982
>6, 79/101, d1=0.428, d2=0.255 g=3.732
>6, 80/101, d1=0.445, d2=0.294 g=3.818
>6, 81/101, d1=0.465, d2=0.341 g=4.639
>6, 82/101, d1=0.484, d2=0.357 g=4.170
>6, 83/101, d1=0.428, d2=0.224 g=3.776
>6, 84/101, d1=0.378, d2=0.279 g=3.762
>6, 85/101, d1=0.432, d2=0.306 g=2.831
>6, 86/101, d1=0.406, d2=0.247 g=2.976
>6, 87/101, d1=0.672, d2=0.578 g=5.630
>6, 88/101, d1=0.627, d2=0.215 g=2.905
>6, 89/101, d1=0.436, d2=0.310 g=5.335
>6, 90/101, d1=0.499, d2=0.227 g=3.105
>6, 91/101, d1=0.398, d2=0.236 g=4.806
>6, 92/101, d1=0.453, d2=0.213 g=2.198
>6, 93/101, d1=0.386, d2=0.526 g=7.028
>6, 94/101, d1=1.186, d2=0.204 g=2.013
>6, 95/101, d1=0.463, d2=0.257 g=5.175
>6, 96/101, d1=0.398, d2=0.226 g=3.903
>6, 97/101, d1=0.513, d2=0.319 g=2.081
>6, 98/101, d1=0.376, d2=0.239 g=4.539
>6, 99/101, d1=0.602, d2=0.234 g=1.103
>6, 100/101, d1=0.438, d2=0.265 g=2.541
>6, 101/101, d1=0.474, d2=0.265 g=2.014
>7, 1/101, d1=0.382, d2=0.351 g=2.092
>7, 2/101, d1=0.409, d2=0.242 g=1.244
>7, 3/101, d1=0.401, d2=0.326 g=1.257
>7, 4/101, d1=0.422, d2=0.199 g=1.308
>7, 5/101, d1=0.428, d2=0.254 g=0.870
>7, 6/101, d1=0.385, d2=0.243 g=1.019
>7, 7/101, d1=0.373, d2=0.219 g=1.175
>7, 8/101, d1=0.527, d2=0.440 g=2.374
>7, 9/101, d1=0.440, d2=0.206 g=1.403
>7, 10/101, d1=0.427, d2=0.222 g=0.353
>7, 11/101, d1=0.369, d2=0.218 g=1.150
>7, 12/101, d1=0.405, d2=0.171 g=0.537
>7, 13/101, d1=0.388, d2=0.263 g=0.210
>7, 14/101, d1=0.376, d2=0.271 g=1.200
>7, 15/101, d1=0.543, d2=0.398 g=1.558
>7, 16/101, d1=0.431, d2=0.225 g=1.049
>7, 17/101, d1=0.383, d2=0.181 g=1.900
>7, 18/101, d1=0.415, d2=0.195 g=0.521
>7, 19/101, d1=0.398, d2=0.272 g=2.192
>7, 20/101, d1=0.431, d2=0.242 g=1.474

>7, 21/101, d1=0.389, d2=0.354 g=2.267
>7, 22/101, d1=0.395, d2=0.263 g=1.010
>7, 23/101, d1=0.400, d2=0.227 g=2.347
>7, 24/101, d1=0.390, d2=0.174 g=2.453
>7, 25/101, d1=0.557, d2=0.481 g=3.660
>7, 26/101, d1=0.368, d2=0.220 g=5.479
>7, 27/101, d1=0.437, d2=0.270 g=3.152
>7, 28/101, d1=0.411, d2=0.237 g=3.789

>7, 29/101, d1=0.382, d2=0.271 g=3.677
>7, 30/101, d1=0.428, d2=0.310 g=3.777
>7, 31/101, d1=0.383, d2=0.209 g=4.009
>7, 32/101, d1=0.395, d2=0.260 g=3.216
>7, 33/101, d1=0.375, d2=0.215 g=3.896
>7, 34/101, d1=0.369, d2=0.261 g=3.865
>7, 35/101, d1=0.455, d2=0.218 g=2.251
>7, 36/101, d1=0.385, d2=0.209 g=2.462
>7, 37/101, d1=0.369, d2=0.248 g=2.041
>7, 38/101, d1=0.368, d2=0.186 g=0.950
>7, 39/101, d1=0.415, d2=0.242 g=0.246
>7, 40/101, d1=0.369, d2=0.270 g=0.950
>7, 41/101, d1=0.365, d2=0.207 g=1.084
>7, 42/101, d1=0.372, d2=0.136 g=0.387
>7, 43/101, d1=0.377, d2=0.223 g=0.281
>7, 44/101, d1=0.372, d2=0.329 g=1.391
>7, 45/101, d1=0.434, d2=0.166 g=0.493
>7, 46/101, d1=0.374, d2=0.293 g=1.945
>7, 47/101, d1=0.360, d2=0.267 g=0.890
>7, 48/101, d1=0.367, d2=0.195 g=0.330
>7, 49/101, d1=0.404, d2=0.238 g=0.256
>7, 50/101, d1=0.360, d2=0.246 g=0.786
>7, 51/101, d1=0.478, d2=0.235 g=0.149
>7, 52/101, d1=0.376, d2=0.294 g=1.477
>7, 53/101, d1=0.456, d2=0.179 g=0.804
>7, 54/101, d1=0.379, d2=0.191 g=0.477
>7, 55/101, d1=0.372, d2=0.185 g=0.847
>7, 56/101, d1=0.375, d2=0.222 g=1.169
>7, 57/101, d1=0.601, d2=1.094 g=8.203
>7, 58/101, d1=1.921, d2=0.191 g=0.281
>7, 59/101, d1=0.511, d2=0.416 g=1.441
>7, 60/101, d1=0.385, d2=0.216 g=2.689
>7, 61/101, d1=0.424, d2=0.167 g=0.921
>7, 62/101, d1=0.369, d2=0.330 g=1.867
>7, 63/101, d1=0.432, d2=0.244 g=1.280
>7, 64/101, d1=0.430, d2=0.297 g=0.694

>7, 65/101, d1=0.400, d2=0.363 g=1.517
>7, 66/101, d1=0.523, d2=0.398 g=2.182
>7, 67/101, d1=0.523, d2=0.279 g=0.764
>7, 68/101, d1=0.597, d2=0.717 g=3.600
>7, 69/101, d1=0.876, d2=0.442 g=1.603
>7, 70/101, d1=0.379, d2=0.258 g=3.235
>7, 71/101, d1=0.697, d2=0.451 g=0.973
>7, 72/101, d1=0.402, d2=0.248 g=1.722
>7, 73/101, d1=0.461, d2=0.286 g=2.045
>7, 74/101, d1=0.480, d2=0.319 g=1.731
>7, 75/101, d1=0.521, d2=0.430 g=2.983
>7, 76/101, d1=0.549, d2=0.244 g=2.197
>7, 77/101, d1=0.395, d2=0.320 g=4.848
>7, 78/101, d1=0.892, d2=0.658 g=4.053
>7, 79/101, d1=0.441, d2=0.269 g=6.784
>7, 80/101, d1=0.549, d2=0.246 g=3.053
>7, 81/101, d1=0.487, d2=0.281 g=4.110
>7, 82/101, d1=0.393, d2=0.186 g=5.125
>7, 83/101, d1=0.386, d2=0.246 g=4.001
>7, 84/101, d1=0.483, d2=0.210 g=0.592
>7, 85/101, d1=0.422, d2=0.429 g=4.309
>7, 86/101, d1=0.564, d2=0.233 g=1.923
>7, 87/101, d1=0.389, d2=0.205 g=0.400
>7, 88/101, d1=0.397, d2=0.268 g=0.480
>7, 89/101, d1=0.391, d2=0.225 g=0.637
>7, 90/101, d1=0.375, d2=0.227 g=0.534
>7, 91/101, d1=0.393, d2=0.168 g=0.285
>7, 92/101, d1=0.408, d2=0.390 g=1.866
>7, 93/101, d1=0.693, d2=0.265 g=0.237
>7, 94/101, d1=0.428, d2=0.258 g=0.965
>7, 95/101, d1=0.397, d2=0.192 g=1.339
>7, 96/101, d1=0.404, d2=0.236 g=0.390
>7, 97/101, d1=0.411, d2=0.344 g=1.135
>7, 98/101, d1=0.477, d2=0.272 g=0.415
>7, 99/101, d1=0.394, d2=0.236 g=0.410
>7, 100/101, d1=0.397, d2=0.282 g=1.007
>7, 101/101, d1=0.359, d2=0.275 g=0.869
>8, 1/101, d1=0.435, d2=0.246 g=0.205
>8, 2/101, d1=0.376, d2=0.273 g=0.868
>8, 3/101, d1=0.431, d2=0.266 g=0.567
>8, 4/101, d1=0.374, d2=0.220 g=1.015
>8, 5/101, d1=0.424, d2=0.174 g=0.174
>8, 6/101, d1=0.390, d2=0.445 g=3.294
>8, 7/101, d1=0.999, d2=0.413 g=0.778
>8, 8/101, d1=0.394, d2=0.136 g=1.406

>8, 9/101, d1=0.388, d2=0.239 g=1.518
>8, 10/101, d1=0.404, d2=0.266 g=1.917
>8, 11/101, d1=0.511, d2=0.419 g=1.471
>8, 12/101, d1=0.431, d2=0.244 g=1.161
>8, 13/101, d1=0.434, d2=0.433 g=1.413
>8, 14/101, d1=0.422, d2=0.175 g=0.901
>8, 15/101, d1=0.426, d2=0.454 g=2.472
>8, 16/101, d1=0.637, d2=0.229 g=0.264
>8, 17/101, d1=0.400, d2=0.382 g=2.082
>8, 18/101, d1=0.583, d2=0.370 g=2.696
>8, 19/101, d1=0.482, d2=0.309 g=2.124
>8, 20/101, d1=0.415, d2=0.371 g=2.924
>8, 21/101, d1=0.463, d2=0.344 g=4.208
>8, 22/101, d1=0.694, d2=0.590 g=4.189
>8, 23/101, d1=0.399, d2=0.215 g=5.590
>8, 24/101, d1=0.476, d2=0.274 g=2.401
>8, 25/101, d1=0.397, d2=0.312 g=4.683
>8, 26/101, d1=0.424, d2=0.209 g=3.500
>8, 27/101, d1=0.503, d2=0.309 g=1.359
>8, 28/101, d1=0.383, d2=0.257 g=1.471
>8, 29/101, d1=0.420, d2=0.239 g=1.377
>8, 30/101, d1=0.382, d2=0.288 g=0.880
>8, 31/101, d1=0.370, d2=0.195 g=0.989
>8, 32/101, d1=0.383, d2=0.246 g=0.260
>8, 33/101, d1=0.389, d2=0.340 g=1.234
>8, 34/101, d1=0.435, d2=0.166 g=1.133
>8, 35/101, d1=0.393, d2=0.214 g=0.478
>8, 36/101, d1=0.379, d2=0.239 g=1.193
>8, 37/101, d1=0.395, d2=0.199 g=0.978
>8, 38/101, d1=0.468, d2=0.297 g=0.156
>8, 39/101, d1=0.366, d2=0.234 g=0.559
>8, 40/101, d1=0.428, d2=0.248 g=0.286
>8, 41/101, d1=0.373, d2=0.250 g=0.301
>8, 42/101, d1=0.405, d2=0.228 g=0.533
>8, 43/101, d1=0.386, d2=0.243 g=0.388
>8, 44/101, d1=0.365, d2=0.186 g=0.499
>8, 45/101, d1=0.408, d2=0.253 g=0.286
>8, 46/101, d1=0.360, d2=0.190 g=1.283
>8, 47/101, d1=0.432, d2=0.266 g=0.425
>8, 48/101, d1=0.384, d2=0.184 g=1.386
>8, 49/101, d1=0.401, d2=0.232 g=0.441
>8, 50/101, d1=0.452, d2=0.403 g=1.393
>8, 51/101, d1=0.463, d2=0.229 g=0.693
>8, 52/101, d1=0.375, d2=0.193 g=1.233
>8, 53/101, d1=0.422, d2=0.246 g=1.421

>8, 54/101, d1=0.467, d2=0.436 g=2.617
>8, 55/101, d1=0.606, d2=0.355 g=2.091
>8, 56/101, d1=0.394, d2=0.305 g=3.625
>8, 57/101, d1=0.533, d2=0.442 g=6.431
>8, 58/101, d1=0.685, d2=0.289 g=2.506
>8, 59/101, d1=0.401, d2=0.315 g=4.732
>8, 60/101, d1=0.596, d2=0.519 g=3.706
>8, 61/101, d1=0.460, d2=0.254 g=2.662
>8, 62/101, d1=0.434, d2=0.586 g=4.743
>8, 63/101, d1=0.542, d2=0.317 g=2.302
>8, 64/101, d1=0.414, d2=0.209 g=1.975
>8, 65/101, d1=0.409, d2=0.206 g=3.614
>8, 66/101, d1=0.743, d2=0.653 g=2.339
>8, 67/101, d1=0.461, d2=0.291 g=2.840
>8, 68/101, d1=0.463, d2=0.241 g=1.651
>8, 69/101, d1=0.393, d2=0.220 g=1.517
>8, 70/101, d1=0.397, d2=0.281 g=1.941
>8, 71/101, d1=0.400, d2=0.140 g=2.448
>8, 72/101, d1=0.405, d2=0.228 g=1.366
>8, 73/101, d1=0.382, d2=0.255 g=1.674
>8, 74/101, d1=0.400, d2=0.170 g=1.369
>8, 75/101, d1=0.417, d2=0.465 g=1.993
>8, 76/101, d1=0.788, d2=0.706 g=2.335
>8, 77/101, d1=0.505, d2=0.235 g=1.072
>8, 78/101, d1=0.412, d2=0.516 g=3.490
>8, 79/101, d1=0.565, d2=0.346 g=3.258
>8, 80/101, d1=0.412, d2=0.212 g=3.011
>8, 81/101, d1=0.504, d2=0.519 g=4.487
>8, 82/101, d1=1.015, d2=0.387 g=1.932
>8, 83/101, d1=0.377, d2=0.355 g=4.420
>8, 84/101, d1=0.534, d2=0.278 g=2.605
>8, 85/101, d1=0.405, d2=0.253 g=3.462
>8, 86/101, d1=0.599, d2=0.576 g=2.569
>8, 87/101, d1=0.400, d2=0.224 g=3.147
>8, 88/101, d1=0.517, d2=0.344 g=1.215
>8, 89/101, d1=0.374, d2=0.282 g=2.828
>8, 90/101, d1=0.495, d2=0.317 g=1.662
>8, 91/101, d1=0.473, d2=0.362 g=3.817
>8, 92/101, d1=0.378, d2=0.231 g=4.527
>8, 93/101, d1=0.989, d2=1.026 g=3.900
>8, 94/101, d1=0.501, d2=0.226 g=3.914
>8, 95/101, d1=0.441, d2=0.299 g=3.361
>8, 96/101, d1=0.389, d2=0.196 g=2.791
>8, 97/101, d1=0.421, d2=0.303 g=2.780
>8, 98/101, d1=0.427, d2=0.308 g=2.647

>8, 99/101, d1=0.447, d2=0.342 g=3.958
>8, 100/101, d1=0.522, d2=0.390 g=4.436
>8, 101/101, d1=0.469, d2=0.290 g=1.816
>9, 1/101, d1=0.408, d2=0.310 g=3.881
>9, 2/101, d1=0.395, d2=0.234 g=3.473
>9, 3/101, d1=0.394, d2=0.260 g=1.999
>9, 4/101, d1=0.393, d2=0.219 g=1.638
>9, 5/101, d1=0.442, d2=0.312 g=2.275
>9, 6/101, d1=0.532, d2=0.374 g=3.748
>9, 7/101, d1=0.487, d2=0.237 g=2.433
>9, 8/101, d1=0.375, d2=0.229 g=1.152
>9, 9/101, d1=0.410, d2=0.250 g=2.112
>9, 10/101, d1=0.369, d2=0.223 g=1.838
>9, 11/101, d1=0.586, d2=0.424 g=1.156
>9, 12/101, d1=0.380, d2=0.188 g=3.223
>9, 13/101, d1=0.482, d2=0.269 g=0.907
>9, 14/101, d1=0.422, d2=0.200 g=1.079
>9, 15/101, d1=0.436, d2=0.266 g=0.657
>9, 16/101, d1=0.447, d2=0.206 g=0.657
>9, 17/101, d1=0.377, d2=0.307 g=1.704
>9, 18/101, d1=0.427, d2=0.268 g=1.875
>9, 19/101, d1=0.416, d2=0.235 g=0.779
>9, 20/101, d1=0.383, d2=0.265 g=3.667
>9, 21/101, d1=0.598, d2=0.245 g=0.627
>9, 22/101, d1=0.416, d2=0.207 g=2.197
>9, 23/101, d1=0.473, d2=0.271 g=0.457
>9, 24/101, d1=0.393, d2=0.199 g=0.487
>9, 25/101, d1=0.378, d2=0.278 g=1.460
>9, 26/101, d1=0.397, d2=0.236 g=1.087
>9, 27/101, d1=0.431, d2=0.206 g=1.103
>9, 28/101, d1=0.585, d2=0.375 g=1.134
>9, 29/101, d1=0.414, d2=0.272 g=2.478
>9, 30/101, d1=0.511, d2=0.450 g=3.003
>9, 31/101, d1=0.468, d2=0.211 g=1.241
>9, 32/101, d1=0.467, d2=0.973 g=5.790
>9, 33/101, d1=1.371, d2=0.257 g=1.283
>9, 34/101, d1=0.467, d2=0.323 g=3.180
>9, 35/101, d1=0.378, d2=0.255 g=3.977
>9, 36/101, d1=0.748, d2=1.106 g=5.510
>9, 37/101, d1=0.683, d2=0.244 g=3.979

>9, 38/101, d1=0.456, d2=0.243 g=3.037
>9, 39/101, d1=0.437, d2=0.253 g=3.599
>9, 40/101, d1=0.447, d2=0.264 g=2.406
>9, 41/101, d1=0.405, d2=0.287 g=4.227

>9, 42/101, d1=0.477, d2=0.309 g=2.990
>9, 43/101, d1=0.387, d2=0.203 g=3.529
>9, 44/101, d1=0.435, d2=0.314 g=3.089
>9, 45/101, d1=0.418, d2=0.175 g=3.668
>9, 46/101, d1=0.414, d2=0.282 g=2.812
>9, 47/101, d1=0.382, d2=0.302 g=3.039
>9, 48/101, d1=0.418, d2=0.271 g=2.979
>9, 49/101, d1=0.517, d2=0.476 g=2.996
>9, 50/101, d1=0.424, d2=0.174 g=1.922
>9, 51/101, d1=0.414, d2=0.303 g=2.348
>9, 52/101, d1=0.420, d2=0.223 g=1.751
>9, 53/101, d1=0.432, d2=0.232 g=1.867
>9, 54/101, d1=0.417, d2=0.289 g=2.213
>9, 55/101, d1=0.457, d2=0.191 g=1.977
>9, 56/101, d1=0.362, d2=0.277 g=1.217
>9, 57/101, d1=0.429, d2=0.425 g=1.912
>9, 58/101, d1=0.461, d2=0.276 g=1.041
>9, 59/101, d1=0.433, d2=0.237 g=1.861
>9, 60/101, d1=0.411, d2=0.234 g=0.807
>9, 61/101, d1=0.413, d2=0.258 g=0.717
>9, 62/101, d1=0.413, d2=0.247 g=0.790
>9, 63/101, d1=0.391, d2=0.267 g=0.760
>9, 64/101, d1=0.394, d2=0.164 g=1.134
>9, 65/101, d1=0.699, d2=0.383 g=0.574
>9, 66/101, d1=0.399, d2=0.232 g=2.681
>9, 67/101, d1=0.440, d2=0.307 g=2.020
>9, 68/101, d1=0.440, d2=0.215 g=1.928
>9, 69/101, d1=0.454, d2=0.362 g=1.684
>9, 70/101, d1=0.466, d2=0.222 g=1.182
>9, 71/101, d1=0.397, d2=0.207 g=1.050
>9, 72/101, d1=0.426, d2=0.261 g=1.360
>9, 73/101, d1=0.387, d2=0.185 g=0.790
>9, 74/101, d1=0.361, d2=0.205 g=1.158
>9, 75/101, d1=0.419, d2=0.330 g=1.500
>9, 76/101, d1=0.417, d2=0.308 g=1.526
>9, 77/101, d1=0.389, d2=0.258 g=1.792
>9, 78/101, d1=0.569, d2=0.648 g=3.541
>9, 79/101, d1=0.617, d2=0.361 g=2.944
>9, 80/101, d1=0.418, d2=0.191 g=3.460
>9, 81/101, d1=0.391, d2=0.239 g=3.661
>9, 82/101, d1=0.507, d2=0.487 g=6.852
>9, 83/101, d1=0.825, d2=0.380 g=3.448
>9, 84/101, d1=0.461, d2=0.273 g=4.514
>9, 85/101, d1=0.380, d2=0.240 g=3.673
>9, 86/101, d1=0.386, d2=0.212 g=4.063

>9, 87/101, d1=0.399, d2=0.282 g=3.213
>9, 88/101, d1=0.431, d2=0.389 g=3.726
>9, 89/101, d1=0.414, d2=0.224 g=3.863
>9, 90/101, d1=0.382, d2=0.189 g=1.823
>9, 91/101, d1=0.429, d2=0.218 g=0.640
>9, 92/101, d1=0.413, d2=0.257 g=1.108
>9, 93/101, d1=0.370, d2=0.322 g=1.050
>9, 94/101, d1=0.366, d2=0.214 g=0.492
>9, 95/101, d1=0.412, d2=0.301 g=0.681
>9, 96/101, d1=0.389, d2=0.194 g=0.887
>9, 97/101, d1=0.371, d2=0.184 g=0.564
>9, 98/101, d1=0.380, d2=0.350 g=1.646
>9, 99/101, d1=0.489, d2=0.223 g=0.556
>9, 100/101, d1=0.393, d2=0.216 g=0.863
>9, 101/101, d1=0.398, d2=0.239 g=0.858

Appendix 8: Images from Unconditional GAN Improv. 1