



GAN: Kako pretvoriti vašeg konja u zebru?

Filip Vesović

Matematička gimnazija

NEDELJA^{v5.0}_{INFORMATIKE}

17. decembar 2018.



Kako pretvoriti vašeg konja u zebru?

Evo ovako:

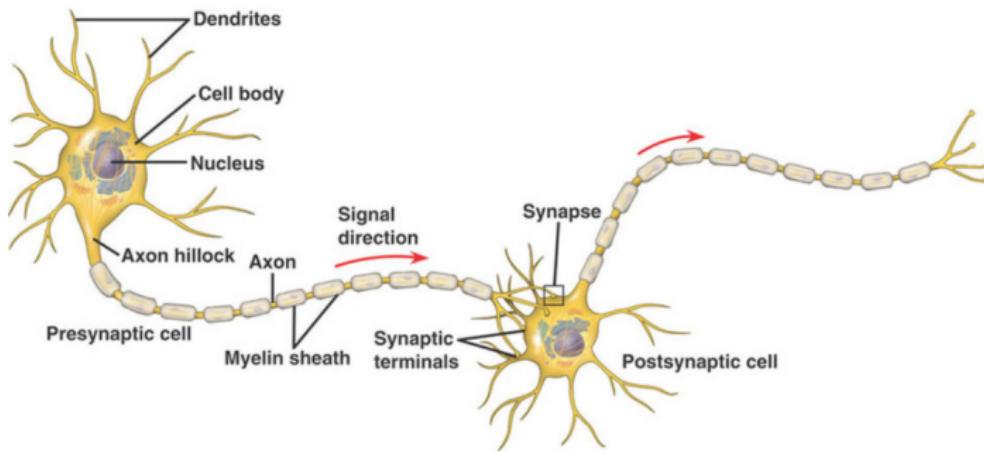




Šta je mašinsko učenje?

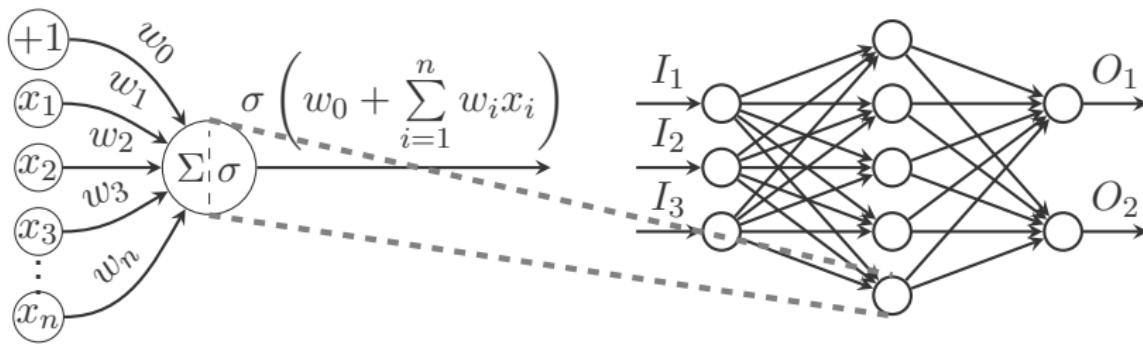
- ▶ Mašinsko učenje - izučava algoritme koji uče iz podataka
- ▶ Neuralna mreža - model baziran na principu rada stvarnog mozga
- ▶ **Suštinska ideja** - izabrati pogodne vrednosti parametara od kojih naša funkcija zavisi tako da ona što bolje aproksimira ciljanu funkciju.

Stvarna neuralna ćelija



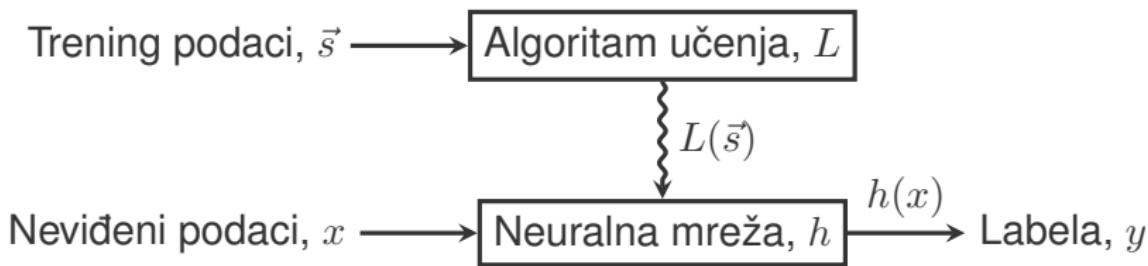
- ▶ Više dendrita preko kojih ćelija prima informaciju
- ▶ Jedan akson kojim predaje informaciju drugoj ćeliji

Veštački neuron i neuralna mreža



- ▶ $\sigma(x)$ - aktivaciona funkcija uvodi nelinearnost u sistem (najčešće ReLU, $\sigma(x) = \max(x, 0)$)
- ▶ **Sloj u neuralnoj mreži** - skup neurona koji dele iste ulazne informacije
- ▶ Više slojeva čini **neuralnu mrežu**

Supervizirano učenje



- ▶ **Trening podatke** - čini skup ulaznih podataka i njihove labele.
- ▶ **Algoritam učenja** - treba da modifikuje neuralnu mrežu tako da se ona dobro ponaša na neviđenim podacima.



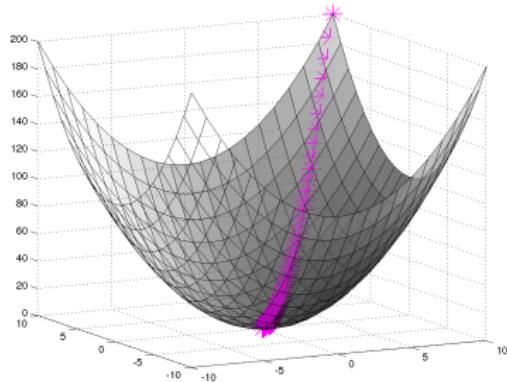
Funkcija greške

- ▶ **Funkcija greške** predstavlja meru koliko izlazi naše mreže odstupaju od željenih.
- ▶ Najčešće se koristi MSE (srednje kvadratne odstojanje) i unakrsna entropija.
- ▶ Cilj algoritma učenja možemo formulisati kao smanjenje očekivane vrednosti funkcije greške.

Gradijentni spust

- ▶ Iterativna metoda za nalazak minimuma funkcije
- ▶ Ideja: ako se nalazimo u tački x , u sledećem koraku pomerićemo se u smeru u kom funkcija najbrže opada.

$$\mathbf{w}_{n+1} = \mathbf{w}_n - \alpha \left(\frac{\partial f}{\partial w_{n1}}, \dots, \frac{\partial f}{\partial w_{nm}} \right)$$

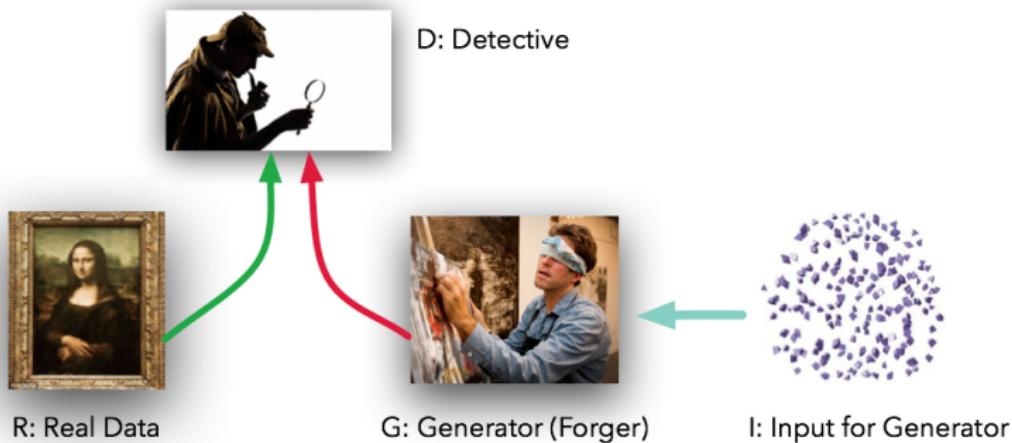




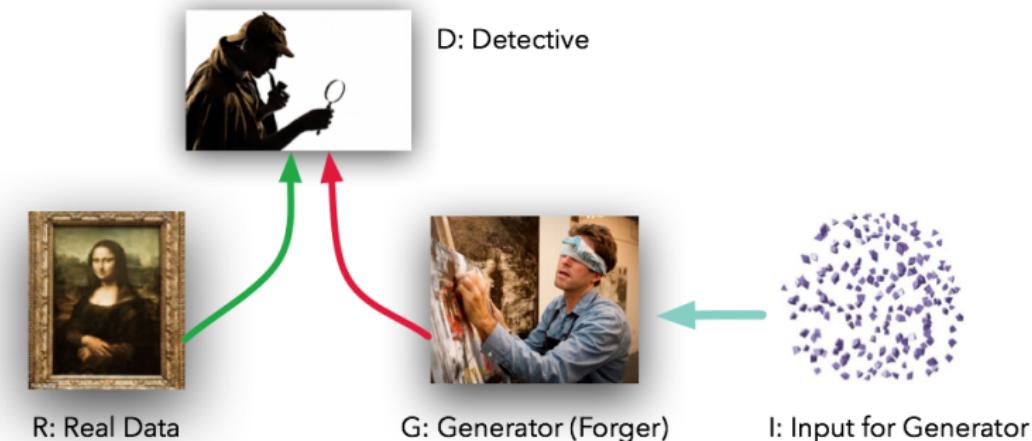
Sta je GAN?

- ▶ **GAN** (Generative Adversarial Networks) - Generativne adverzalne mreže
- ▶ Dve neuralne mreže koje se međusobno takmiče (zero-sum game).
- ▶ Prva mreža (generator) ulaz transformiše tako da liči na podatke iz trening skupa, a druga mreža (diskriminator) odlučuje da li je podatak pravi ili lažan.

Arhitektura GAN



Arhitektura GAN



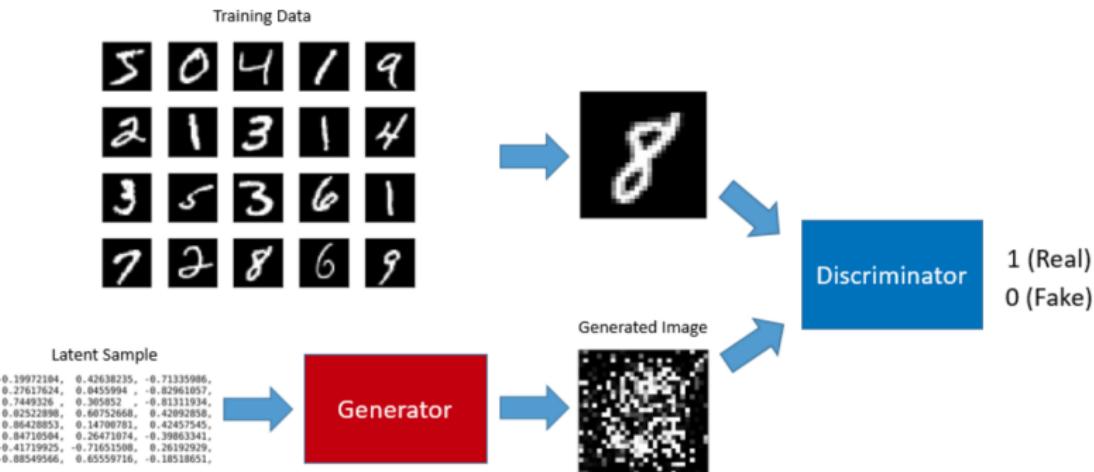
- ▶ Cilj generatora (falsifikatora) je da nauči što bolje da kreira podatke nalik stvarnim podacima.
- ▶ Cilj diskriminatora (detektiva) je da nauči što bolje da otkrije koji podatak je pravi, a koji je generisan.



Arhitektura GAN

- ▶ Na ulaz generatora dovodimo $z \sim p_z(z)$, a izlaz generatora označićemo sa $G(z)$. Generator transformiše ulaznu raspodelu u novu raspodelu $G(z) \sim p_g$.
- ▶ Na ulaz diskriminatora dovodimo ili izlaz generatora $G(z)$ ili pravi podatak iz raspodele p_d . Izlaz diskriminatora treba da nam kaže verovatnoću da je na ulaz doveden pravi podatak, a ne lažni (generisani).
- ▶ Diskriminator treba da utvrdi da li podatak dolazi iz raspodele p_g ili p_d .

Arhitektura GAN



Treniranje GAN



Diskriminat i generator igraju sledeću igru:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_d(x)} [\log D(x)] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))]$$



Treniranje GAN

- ▶ Imamo dve faze treniranja: treniranje generatora, i treniranje diskriminatora
- ▶ **Treniranje generatora** - parametre diskriminatora ne ažuriramo, već ih posmatramo kao fiksne.
- ▶ **Treniranje diskriminatora** - obrnuto, generator se posmatra kao fiksan.



Treniranje GAN

for number of training iterations **do**

for k steps **do**

- Sample minibatch of m noise samples $\{\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(m)}\}$ from noise prior $p_g(\mathbf{z})$.
- Sample minibatch of m examples $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\mathbf{x})$.
- Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^m \left[\log D(\mathbf{x}^{(i)}) + \log (1 - D(G(\mathbf{z}^{(i)}))) \right].$$

end for

- Sample minibatch of m noise samples $\{\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(m)}\}$ from noise prior $p_g(\mathbf{z})$.
- Update the generator by descending its stochastic gradient:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^m \log (1 - D(G(\mathbf{z}^{(i)}))).$$

end for



Suština iza GANova

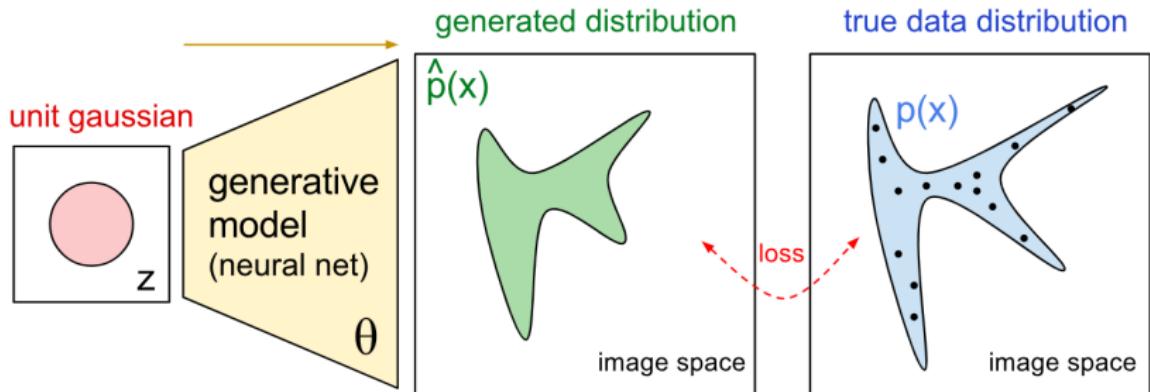
- ▶ Treniranjem običnih neuralnih mreža, mi minimizujemo funkciju greške u nadi da će naša neuralna mreža uspeti da generalizuje.
- ▶ Postavlja se pitanje kako odabratи dobru funkciju greške?
- ▶ Zašto ne bismo funkciju greške zamenili neuralnom mrežom koja će da nauči dobru funkciju greške?
- ▶ \implies GAN



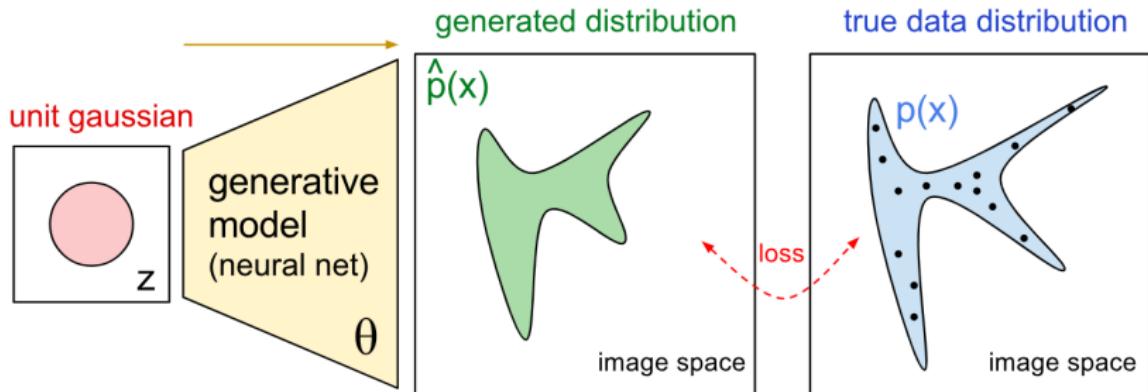
Entropija

- ▶ $H(X) = - \sum_{i=1}^n P(x_i) \log(P(x_i)).$
- ▶ Za bazu logaritma se najčešće uzima 2, 10 ili e.
- ▶ Može se shvatiti kao prosečna količina informacija koju neki random proces stvara.
- ▶ Entropija fer novčića ($p = 1/2$) je 1, a entropija novćica koji uvek pada na istu stranu ($p = 0$) je 0.

Žašto GANovi rade?



Žašto GANovi rade?



- ▶ Kako izmeriti rastojanje između dve raspodele?



Žašto GANovi rade?

Neki od mogućih načina:

- ▶ KL divergence
- ▶ JS divergence
- ▶ Earth mover distanca
- ▶ ..



Kullback–Leibler divergencija

- ▶ KL divergencija:

$$D_{KL}(P\|Q) = - \sum_i P(i) \log \left(\frac{Q(i)}{P(i)} \right)$$

- ▶ U teoriji kodiranja, očekivan broj dodatnih bitova potrebnih da kodiramo uzorke iz P koristeći kod optimizovan za Q umesto P
- ▶ Ako su P i Q iste raspodele $\implies D_{KL}(P\|Q) = 0$
- ▶ Nije simetrična

$$D_{KL}(P\|Q) \neq D_{KL}(Q\|P)$$

C



Jensen-Shannon divergencija

- ▶ JS divergencija:

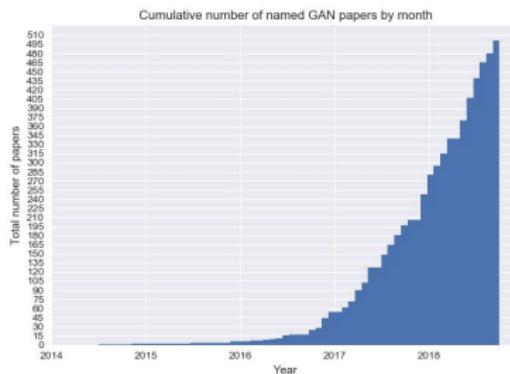
$$JSD_{KL}(P\|Q) = \frac{1}{2}D_{KL}(P\|M) + \frac{1}{2}D_{KL}(Q\|M)$$

gde je $M = \frac{1}{2}(P + Q)$

- ▶ Dokazuje se da GAN koji smo danas uveli minimizuje JS divergenciju.

Malo istorije

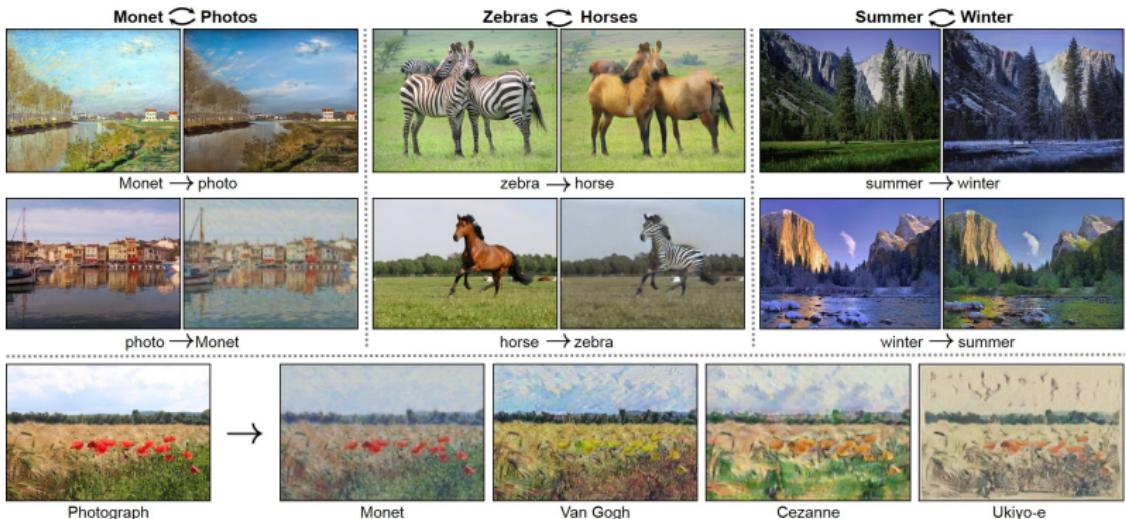
- ▶ Rad objavljen 2014. godine od strane Ian Goodfellow et al.
- ▶ Ideja za GANove nastala posle par pića u pubu.



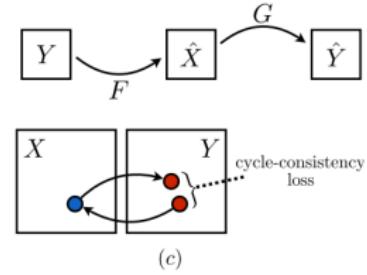
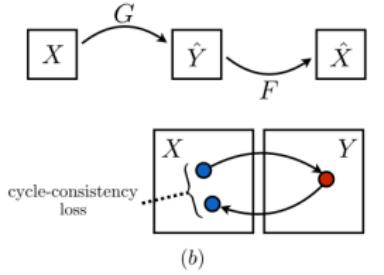
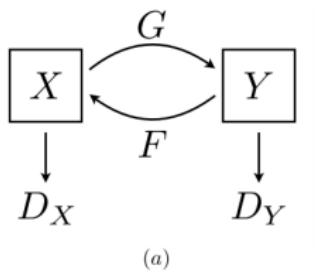
Slika: Ukupan broj GAN papera kroz mesecce

- ▶ Od tada nastale mnoge varijante kao što su DCGAN, WGAN, CycleGAN..

CycleGAN



CycleGAN

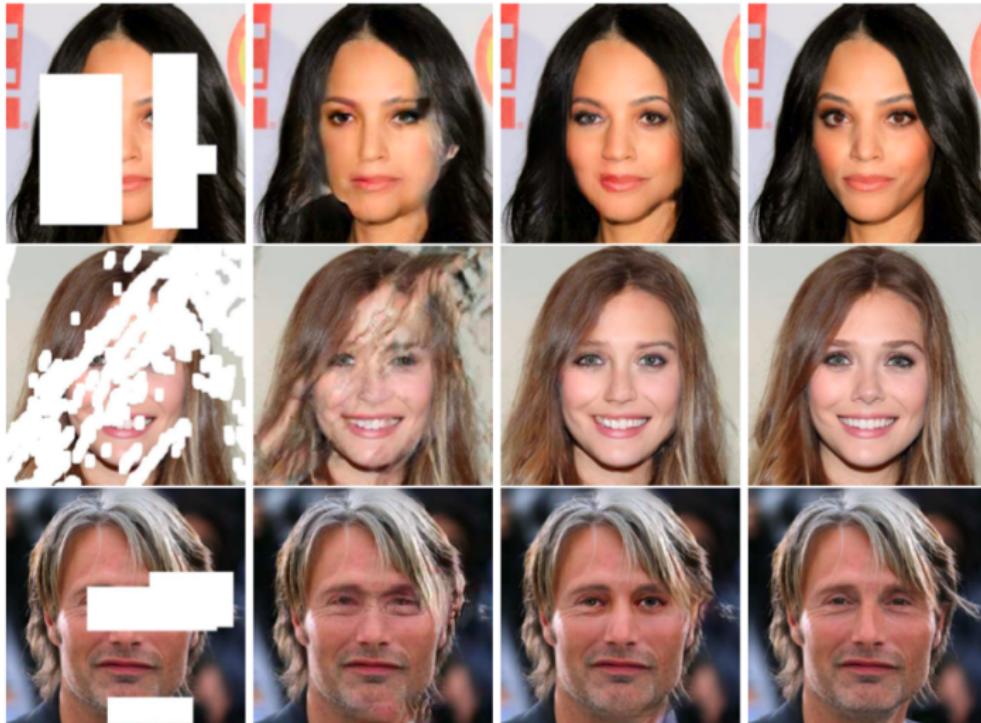


Fake faces



Figure 5: 1024×1024 images generated using the CELEBA-HQ dataset. See Appendix F for a larger set of results, and the accompanying video for latent space interpolations. On the right, two images from an earlier megapixel GAN by Marchesi (2017) show limited detail and variation.

Inprinting



(a) Input

(b) GntIpt

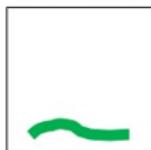
(c) PConv(Ours)

(d) Ground Truth



Interactive GAN

User edits



Generated images



- Color
- - - Sketch

Interpolation

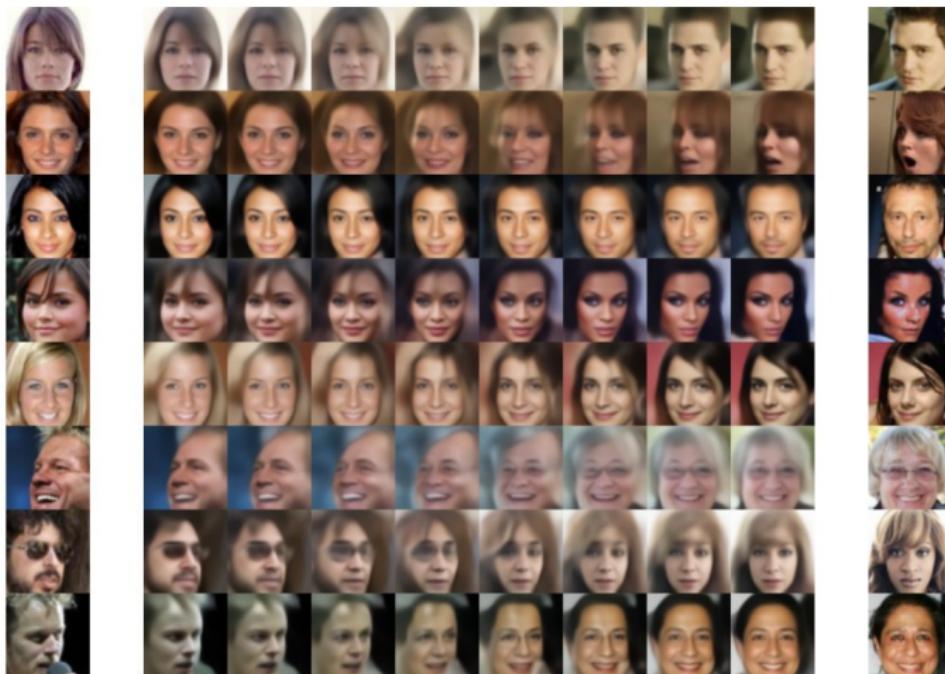
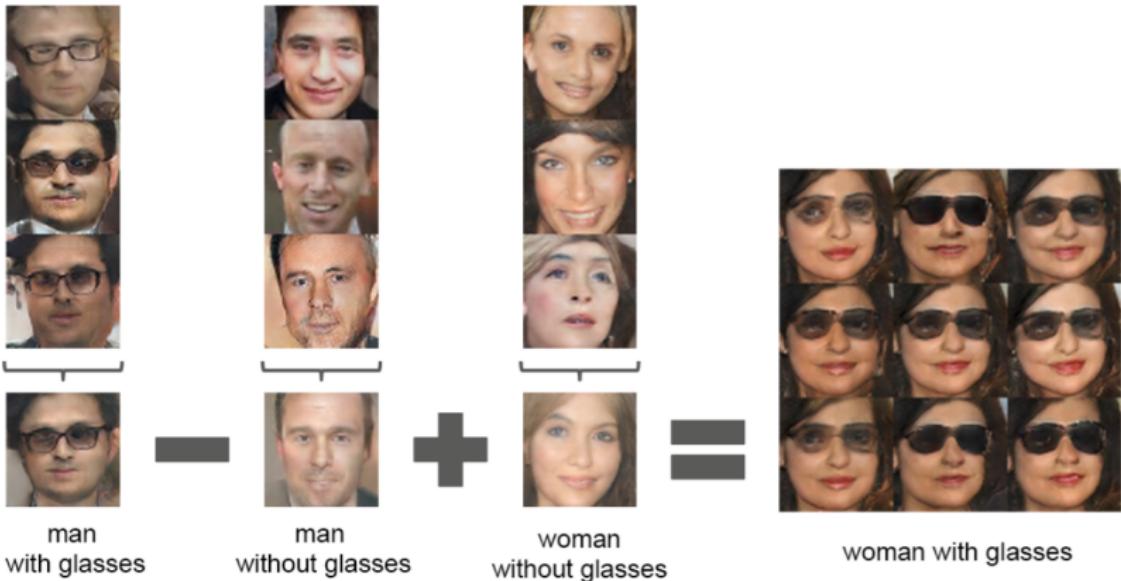


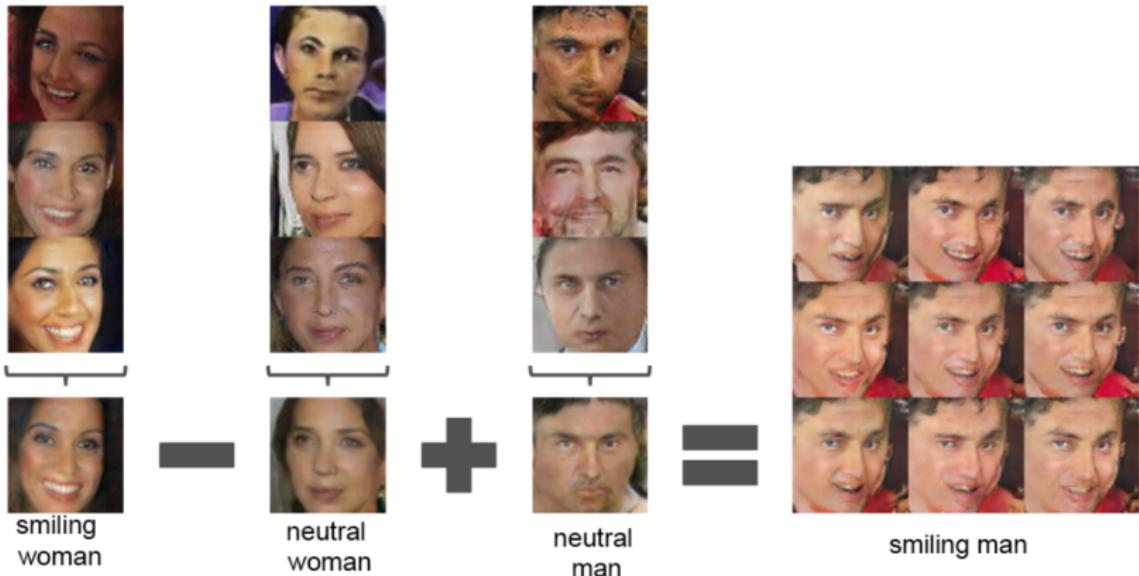
Figure 6. Interpolation experiments for celebA

Latent space arithmetic





Latent space arithmetic



Q&A



Pitanja?