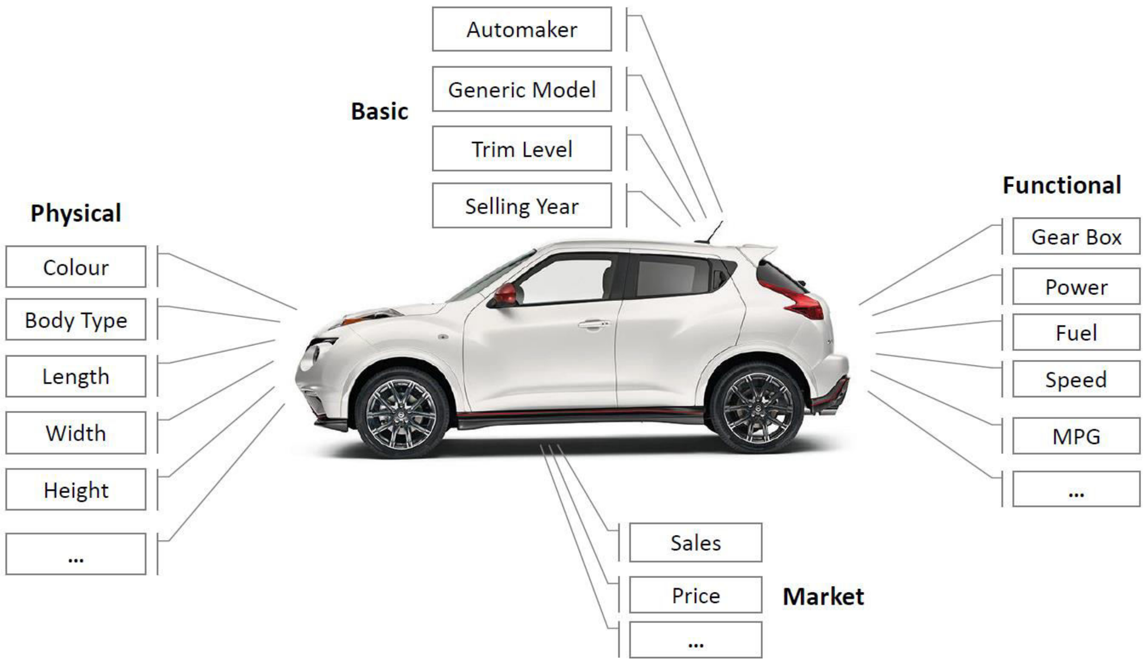
Representation Learning: Compositionality and Disentanglement

Representations as Attributes

- The representations we described so far are quite abstract
- Evaluated by downstream task performance
- Here, we think of representations as consisting of attributes



Compositionality in ML

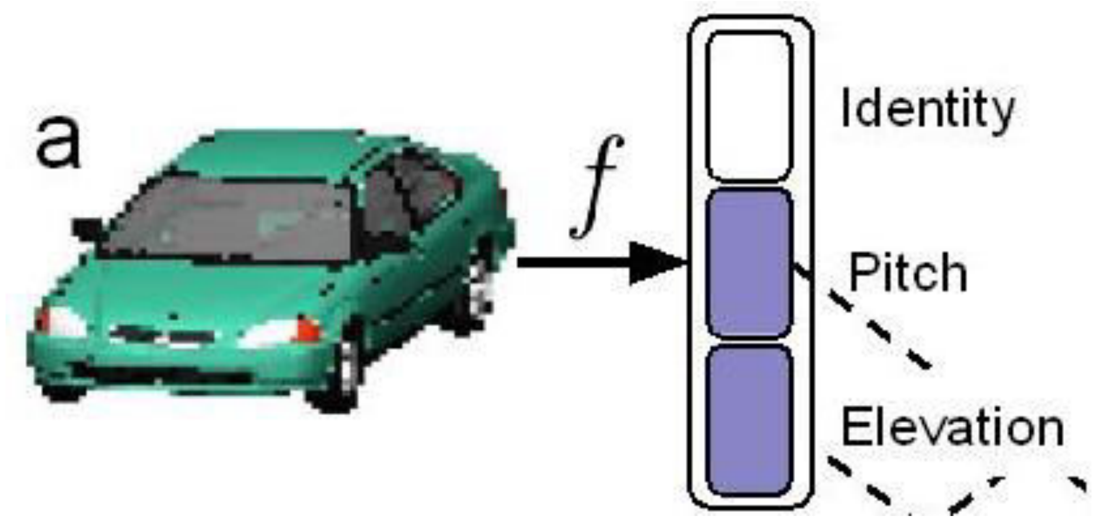
- Given a dataset where each image has two labels, fruit type and color
- Assume in the dataset we see either red apples or yellow bananas
- At test, we see a yellow apple
- We ask: “what fruit is it?”

Ways to Overcome It

- Need more information – different methods for injecting it:
  - Option 1: biasing architecture to differentiate between attributes
  - Option 2: reducing bias e.g. seeing many (color, fruit) combinations

Disentanglement

- Disentangled representation:
  - Each dimension is informative over at most a single attribute
  - Every attribute is predictable from the representation



Disentanglement Entails Compositionality

- Disentanglement is harder compositionality – entails it
- Trivial: every attribute is representated by different dimensions
- Compositionality: problem biased datasets
- Disentanglement: problem even in unbiased datasets

Disentanglement Entails Classification

- Disentanglement is harder than classification
- Trivial: every attribute is represented spearately

Quest for Unsupervised Disentanglement

- Unsupervised disentangled representations – holy grail of SSL
- This is probably impossible
- Assumes we get a bunch of unlabeled images and classify all attributes without supervision – too good to be true

Identifibility in the Linear Setting

- Assume we have two attributes x_1, x_2 – which are not observed
- Pass through a linear generative process G – physics of the world
- G is invertible by unknown
- Observe

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = G \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

- Can we identify G and thus recover the true x_1, x_2?

Independent Component Analysis (ICA)

- ICA observes that x_1,x_2 unidentifiable if they are Gaussian (Why?)
- Instead, it assumes x_1,x_2 are highly non-Gaussian, independent
- Main idea – combination of non Gaussians is less non-Gaussian
- Greedily recovers the most non-Gaussian combination of y_1, y_2

$$min_v \sum_i \rho(v \cdot x_i)$$

- Different non-Gaussianity measures can be used

Nonlinear Identifiability Results

- There is a body of theory examining when attributes are recoverable
- $(y_1, y_2, y_3, ...) = G(x_1, x_2, x_3, ...)$
- G non-linear, x and G are unknown
- Indentifiability guranteed only in **very** limited settings
- Out of scope for this course – very interesting if you like maths!

BetaVAE for Unsupervised Disentanglement

- BetaVAE: normal VAE but with larger weight (beta) on the KL term

$$L_{\mathrm{BETA}}(\phi, \beta) = -\mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) + \beta D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}))$$

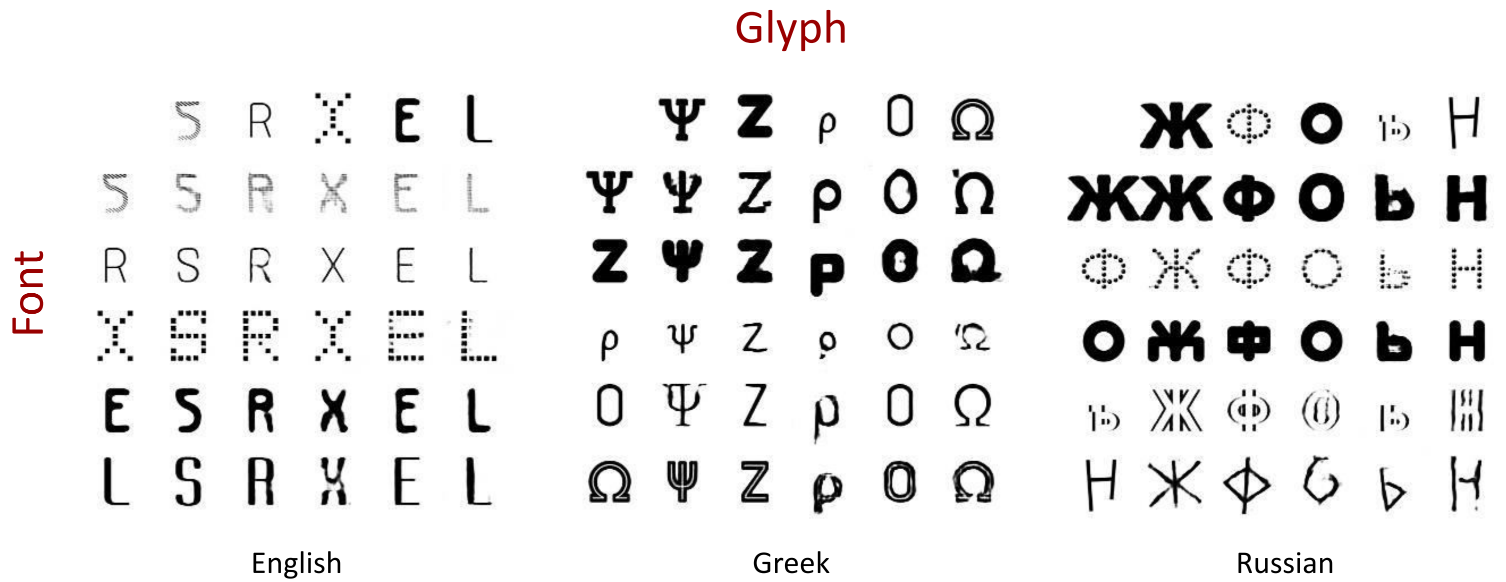
- Intuition: makes the latent codes more Gaussian
- More Gaussian means more independent
- Locatello et al. showed this does not work in general (why?)

How to Evaluate Disentanglement - DCI

- Latent code with 10 dims
- Assume there are K factors of variation
- Expect: one latent code for every factors, 10 – K empty codes
- DCI metrics:
    - Completeness – each factor described by at least 1 code dim
    - Disentanglement – each code dim correlated with at most 1 factor
    - Informaticeness – all factors are described by code

Conditional Disentanglement

- Unsupervised disentanglement may be too hard
- Let’s tackle a different setting – conditional disentanglement
- Every image x is also labeled with its condition c
- This can really be any attribute e.g. pose, car model, painting/photo



VAE for Conditional Disentanglement

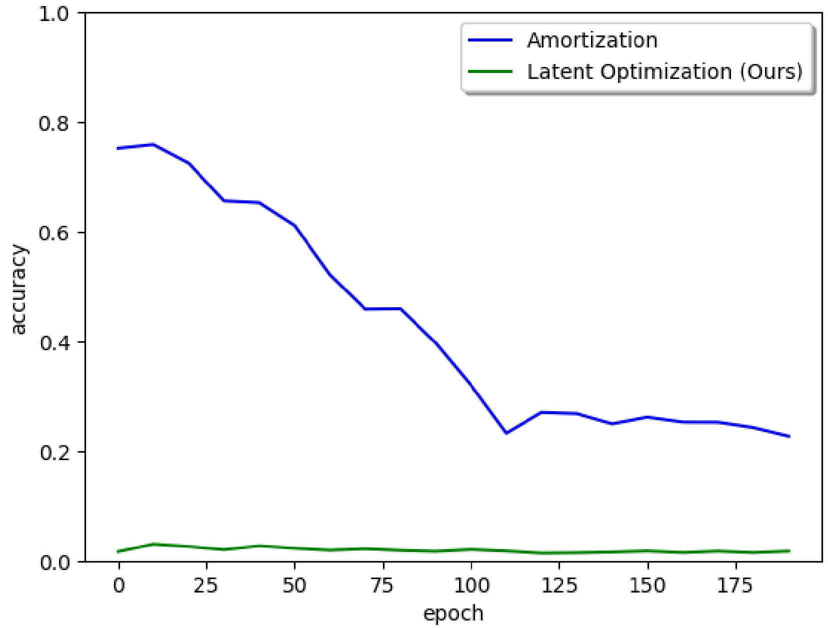
- Choose a prior such that p(z|c) = N(0, I)
- Reminder – the ELBO is given by:

$$L(S,G) = \sum_{(x,c)} E_{z\sim q_{x,c}}||x - G(c,\, z)||^2 + KL(q_{x,c}||p(z|c)) + \log(p(c))$$

- As p(z) = sum_c p(z|c)p(c) = p(z|c), z does not depend on c
- The combination of z, c must represent all attributes in x
- Independence + completeness -> z includes all attributes but x

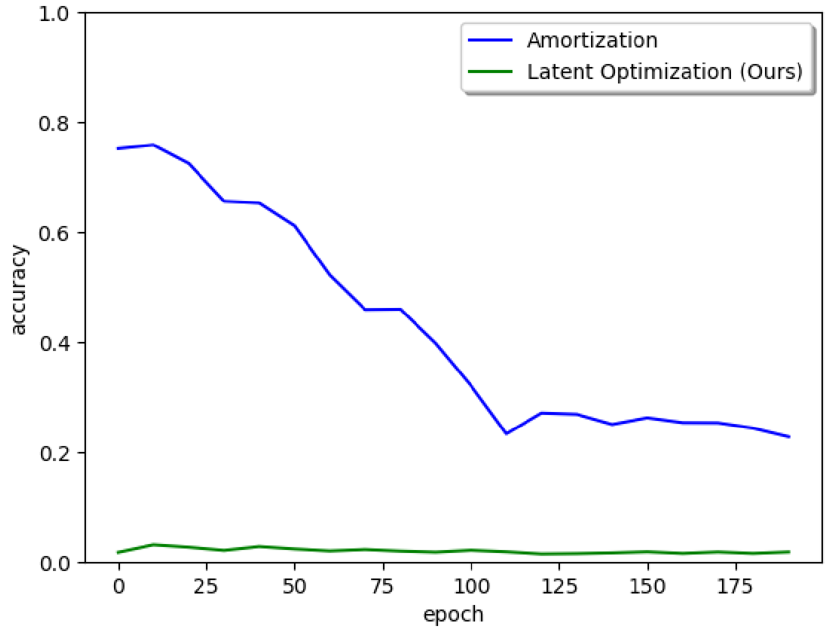
Amortized VAE: Bad for Disentanglement

- The core idea in VAE was using an encoder to predict q(z|x) given x
- Consider what happens at initilization, encoder has random weights
- Latent code contains random combination of attributes – entangled
- While training loss enforces disentanglement, z does not recover



LORD: Latent Optimization is King

- For each image: optimize the expectation mu_x of p(z|x) = N(mu_x, s)
- As each mu_x is initialized randomly, independent of c
- During training mu becomes more informative on x, but still not on c



Does This Solve Disentanglement?

- Identifiably of the unknown attributes is still **not** guranteed!
- This is not hard to see: for images of (apples, bananas), (red, yellow)
- Every image is tagged with color (c) but not fruit
- LORD ensures that (z, c) describe all attributes, and z,c independent
- Both options are feasible solutions, but only the first is helpful

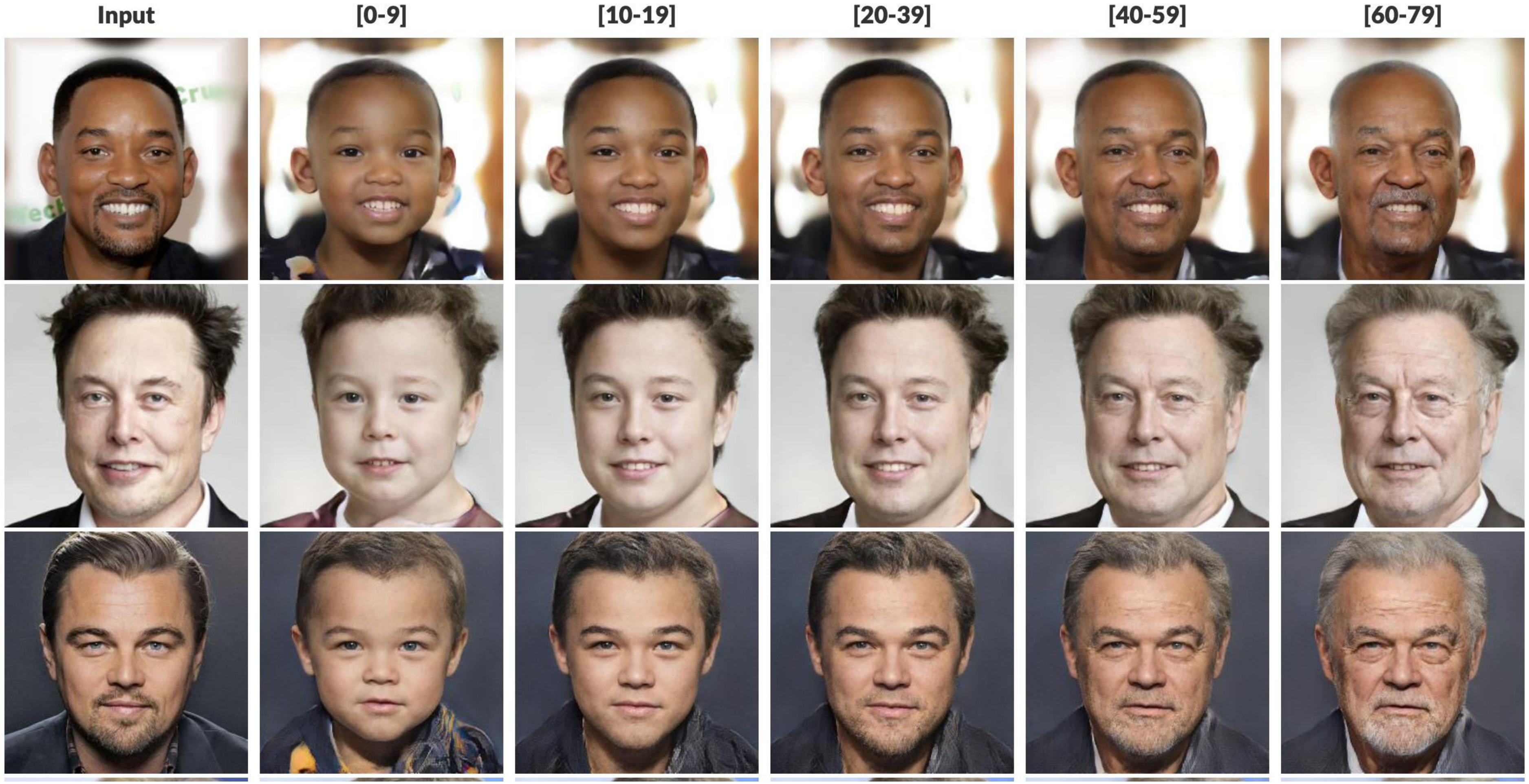
	c = red	c = yellow
z=0	apples	apples
z=1	bananas	bananas

	c= red	c = yellow
z=0	apples	bananas
z=1	bananas	apples

Why Does LORD Work in Practice?

- Inductive bias - magic of CNNs
- While multiple solutions are feasible they prefer the correct one
- This is clearly not **always** going to be true!
- Occurs in many interesting cases though
- https://github.com/avivga/lord-pytorch

Age Transfer using LORD



Benefits of Correct Disentanglement

- Non-disentangled representations mix different conditions (species)