Concept-based explanations

CSEP 590B: Explainable AI

Hugh Chen, Ian Covert & Su-In Lee
University of Washington

©2022 Su-In Lee

Course announcements

- HW1 grades are released
- We need one more week 8 discussion leader

©2022 Su-In Lee

Recall: decomposability

- Do model components have an intuitive role (inputs, parameters, calculations)?
  - Examples: splits in decision tree, weights in linear model, **input features**

- **Concept explanations** consider the role of *high-level concepts* rather than original inputs
  - Potentially more intuitive, meaningful to humans

Lipton, "The Mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery" (2018)

©2022 Su-In Lee

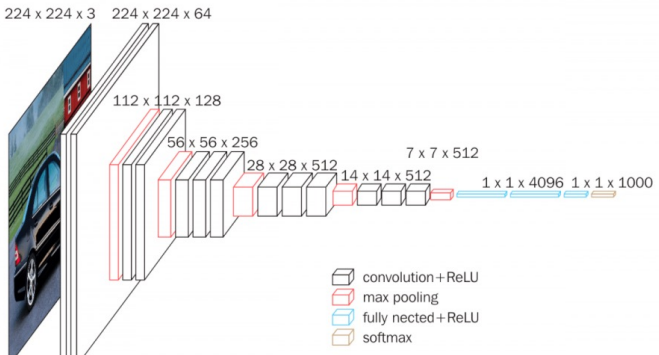
Setup

- Focusing on high-dimensional data
  - Mainly images, possibly genomics or NLP
- With high-dimensional data, humans may prefer to operate on high-level concepts
  - A processed version of input, possibly with fewer dimensions (compressed)
  - More intuitive meaning, more direct relationship with outcome than original features (e.g., pixels)

©2022 Su-In Lee

High-level features in DNNs

- Conventional wisdom about how DNNs process images:
  - Input layer is pixels
  - First layer detects edges
  - Next layers find parts
  - Highest layers detect objects
  - Last layer makes classification



Zeiler & Fergus, "Visualizing and understanding convolutional networks" (2013)

©2022 Su-In Lee

Analogy to human reasoning

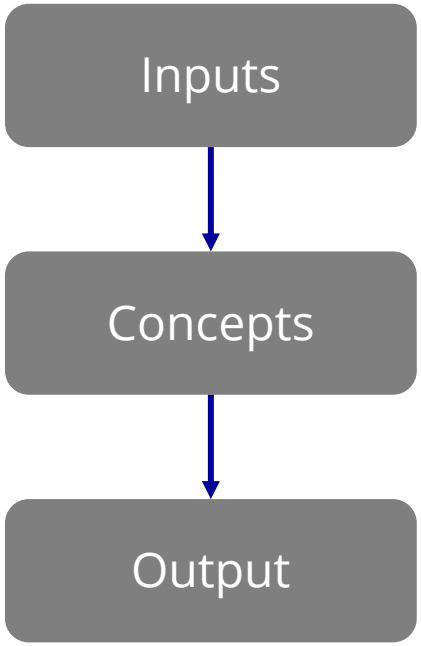
- Seemingly true for DNNs, and interesting to compare with humans
  - Humans seem to reason in a similar, hierarchical manner
  - Typically prefer explanations based on high-level, intuitive concepts

- Can we incorporate this into an explanation approach?

©2022 Su-In Lee

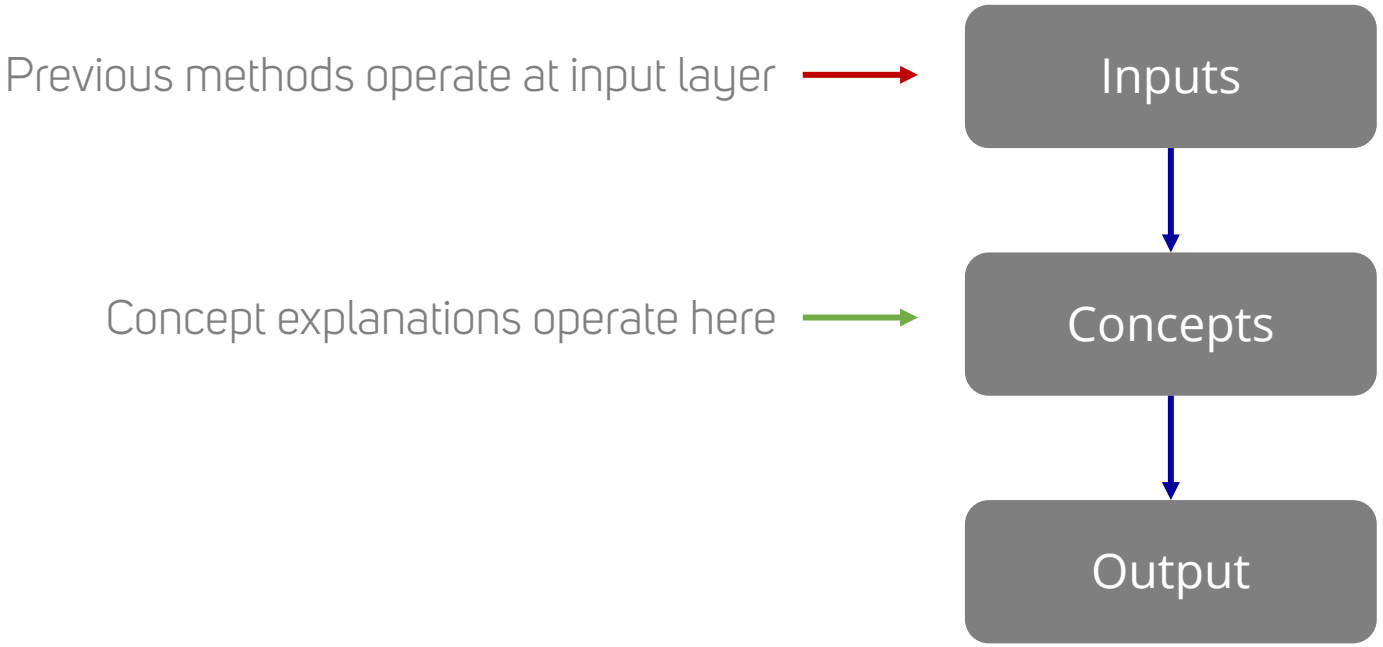
Concept representation

- Consider concepts as an intermediate representation
  - Examples: color, texture, object parts, shape
- Properties:
  - Compressed (fewer dimensions)
  - Sacrifices minimal information
  - Intuitive meaning
  - Simpler relationship with output



©2022 Su-In Lee

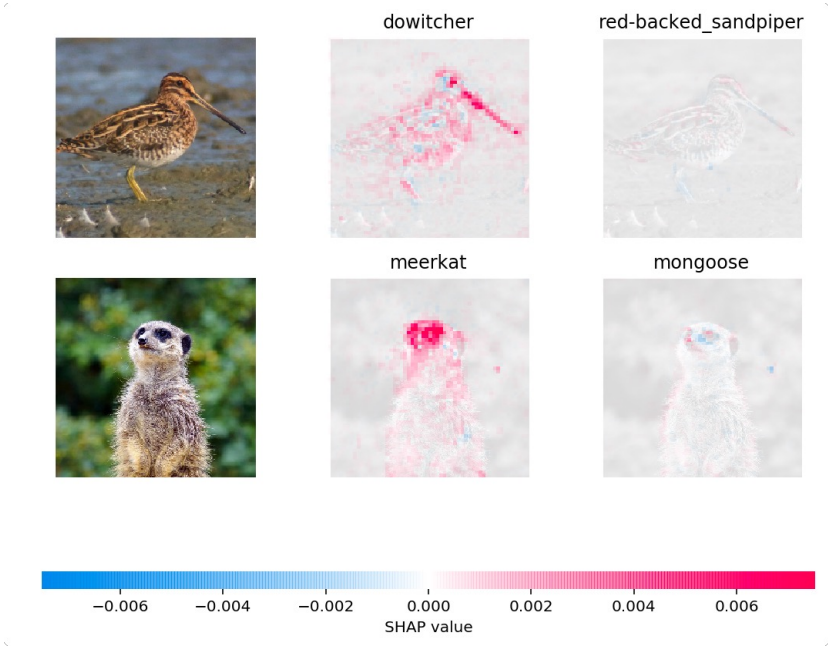
Concept explanations



©2022 Su-In Lee

Image example

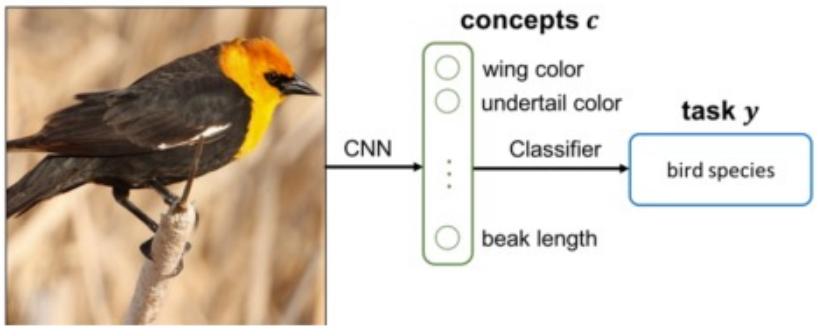
- Explaining at pixel-level localizes important information
  - But is importance due to color, texture, shape, or something else?



©2022 Su-In Lee

Image example (cont.)

- Alternatively, explanations can be based on high-level concepts
- Potentially more informative, intuitive for humans



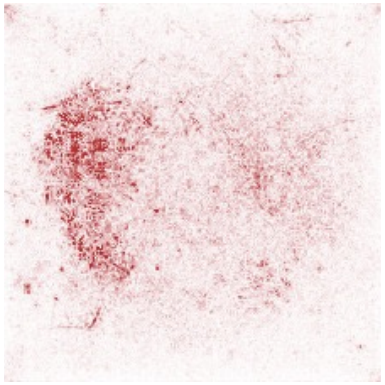
©2022 Su-In Lee

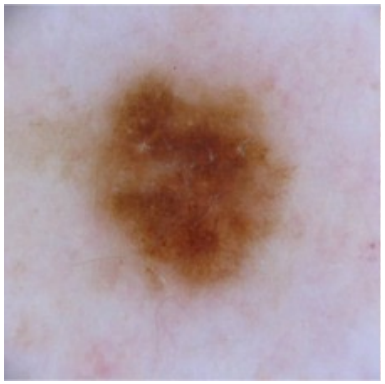
Medical image example

Input image

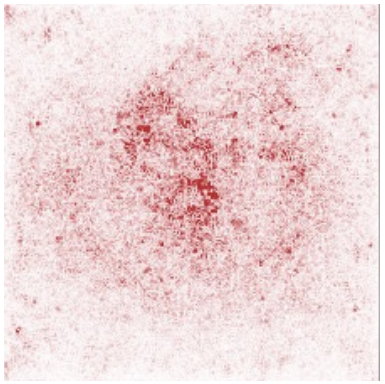
Benign

Saliency map





Benign



Can we go beyond localization?

Provided by Alex DeGrave, MD/PhD student in the AIMS lab

©2022 Su-In Lee

Challenges

- Which concepts should we consider?
- How do we obtain a concept-based representation of the input data?

- Possible approaches:
  - Adjust the model to guarantee that specific concepts are used
  - Use a standard model, then discover how concepts are represented within the model

©2022 Su-In Lee

Today

- Section 1
  - Concept bottleneck models 
  - Concept activation vectors
  - StylEx
- Section 2
  - Neuron interpretation

©2022 Su-In Lee

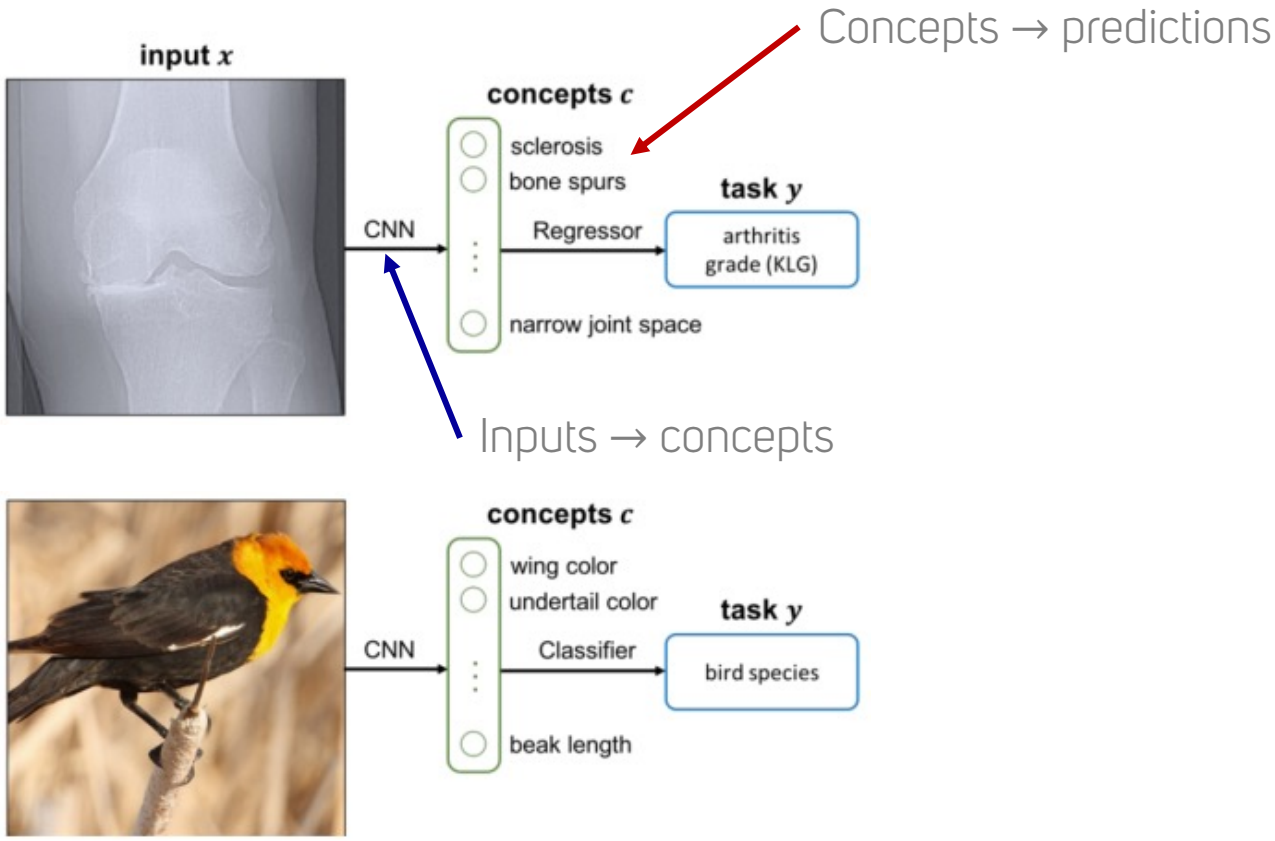
Main idea

- Force a deep learning model to represent specific concepts before making prediction
- Then, use intermediate concept representation to understand the model's dependencies

Koh et al., "Concept bottleneck models" (2020)

©2022 Su-In Lee

Concept bottleneck models



Koh et al., "Concept bottleneck models" (2020)

©2022 Su-In Lee

Learning concept bottleneck models

- Training data $\left\{\left(x^{(i)}, y^{(i)}, c^{(i)}\right)\right\}_{i=1}^{n}$, where x is input, y is label, and c is **concept vector**
- Create an architecture with bottleneck layer
  - Map from inputs to concepts with $\hat{c} = g(x)$
  - Then map to labels with $f\left(g(x)\right)$
- Train the model to accurately predict both concepts and labels
  - Can train either jointly or sequentially

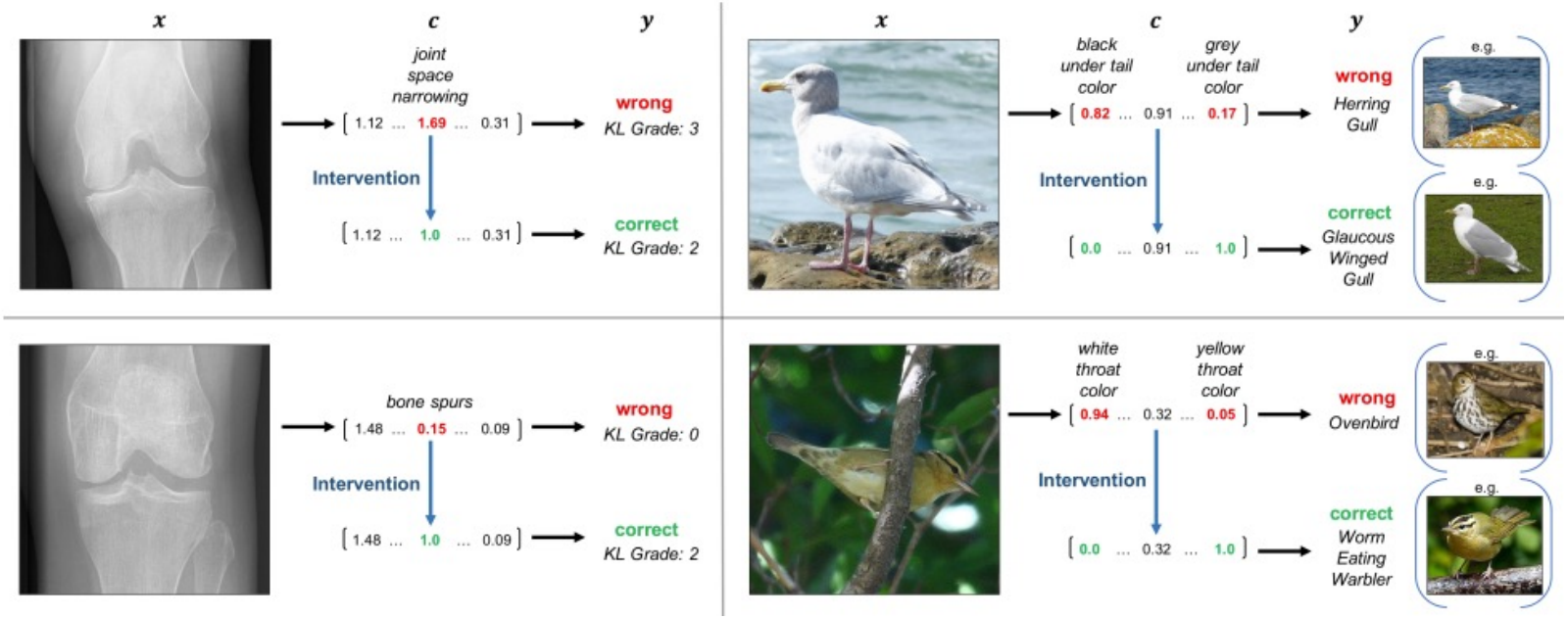
©2022 Su-In Lee

Test-time interventions

- Analyze how the model responds to changes in the predicted concepts
- Intervene on samples by replacing incorrectly predicted concepts with true concept values

©2022 Su-In Lee

Successful test-time interventions



Intervening on one or more concepts can correct the model prediction

©2022 Su-In Lee

Generating explanations

- Additionally, we can apply explanation approaches from previous lectures
- **Gradient-based explanations:**
  - Is the output sensitive to a concept being slightly more expressed?
- **Removal-based explanations:**
  - Is the output sensitive to removing information from one or more concepts?
  - E.g., leave-one-out or Shapley values
- **Counterfactual explanations** (next time)

©2022 Su-In Lee

Remarks

- **Pros:**
  - CBM ensures the model operates on a known set of concepts (and nothing else)
  - Enables intervention and explanation via concepts

- **Cons:**
  - Must use modified architecture
  - Requires comprehensive set of concepts for high accuracy
  - Requires concept annotations in training data

©2022 Su-In Lee

Today

- Section 1
  - Concept bottleneck models
  - Concept activation vectors 
  - StylEx
- Section 2
  - Neuron interpretation

©2022 Su-In Lee

Main idea

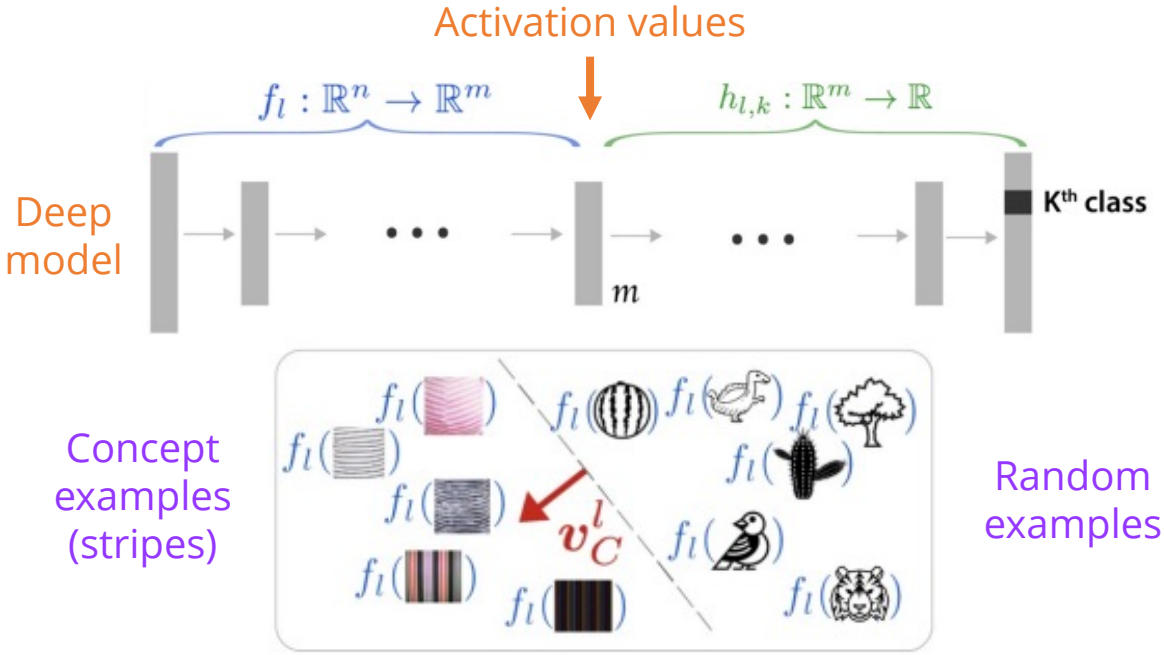
- Post-hoc approach to identify concepts in a model's latent space (internal representation)
  - Alternative to using a concept bottleneck layer
- After training the model, use concept samples to find **concept activation vectors (CAV)**
- Investigate a prediction's sensitivity to concepts

Kim et al. "Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV)" (2018)

©2022 Su-In Lee

Concept activation vector (CAV)

- Choose a concept, select a hidden layer
- Find the direction separating samples that represent the concept



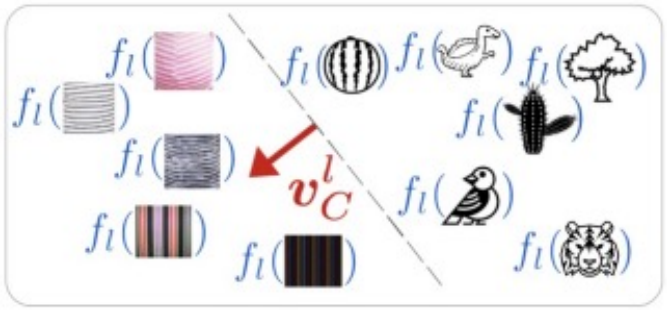
CAV based on linear classifier

©2022 Su-In Lee

CAV computation

- Calculate embeddings for positive and negative concept examples
- Train a linear classifier to separate them
- CAV is vector orthogonal to classification boundary

Concept examples (stripes)



Random examples

CAV based on linear classifier

©2022 Su-In Lee

Sanity checks

- Calculate CAV for a given concept
- Examine images strongly activated along CAV direction

top 3 images of salmon similar to striped concept

bottom 3 images of salmon similar to striped concept

top 3 images of corgis similar to knitted concept

bottom 3 images of corgis similar to knitted concept

©2022 Su-In Lee

Conceptual sensitivity

- Recall, input gradients consider sensitivity to small changes in pixel intensity

- Here, **conceptual sensitivity** is about small changes in a *concept*'s intensity
  - Calculate the impact of small perturbations in CAV direction
  - Equivalent to a directional derivative

©2022 Su-In Lee

Conceptual sensitivity (cont.)

- Let x be an input, k class of interest
- Let $f_l(x)$ be intermediate representation and $h_{l,k}\big(f_l(x)\big)$ the prediction for class y
- Let v_C^l be the CAV for concept C
- Conceptual sensitivity $S_{C,k,l}(x) \in \mathbb{R}$ is given by:

$$S_{C,k,l}(x) = \lim_{\epsilon \to 0} \frac{h_{l,k}\big(f_l(x) + \epsilon v_C^l\big) - h_{l,k}\big(f_l(x)\big)}{\epsilon}$$

Directional derivative

$$= \nabla h_{l,k}\big(f_l(x)\big) \cdot v_C^l$$

Can be obtained via dot product

©2022 Su-In Lee

Conceptual sensitivity (cont.)

- Conceptual sensitivity:

$$S_{C,k,l}(x) = \nabla h_{l,k}\left(f_l(x)\right) \cdot v_C^l$$

Sample

Conceptual sensitivity $S_{C,k,l}($  $)$

Directional derivative $= \nabla h_{l,k}\left(f_l($  $)\right) \cdot v_C^l$

CAV (e.g., stripes)

Output function

Embedding function

©2022 Su-In Lee

Local explanations

- Consider input x, class of interest k
- How relevant is each concept to this prediction?
- We can calculate conceptual sensitivity $S_{C,k,l}(x)$ for all concepts C

©2022 Su-In Lee

Global explanations

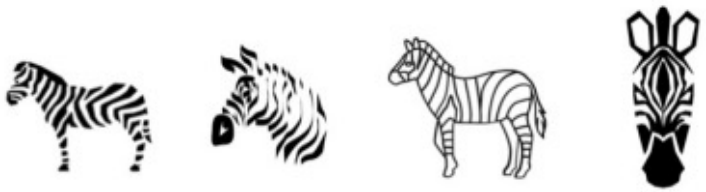
- Consider a class of interest k, and a concept C
- How relevant is the concept to this class?
- Kim et al. propose the **TCAV score** to summarize many local explanations:

$$\mathrm{TCAV}_{C,k,l} = \frac{\left|\{x \in X_k : S_{C,k,l}(x) > 0\}\right|}{|X_k|}$$

Set of examples with class k

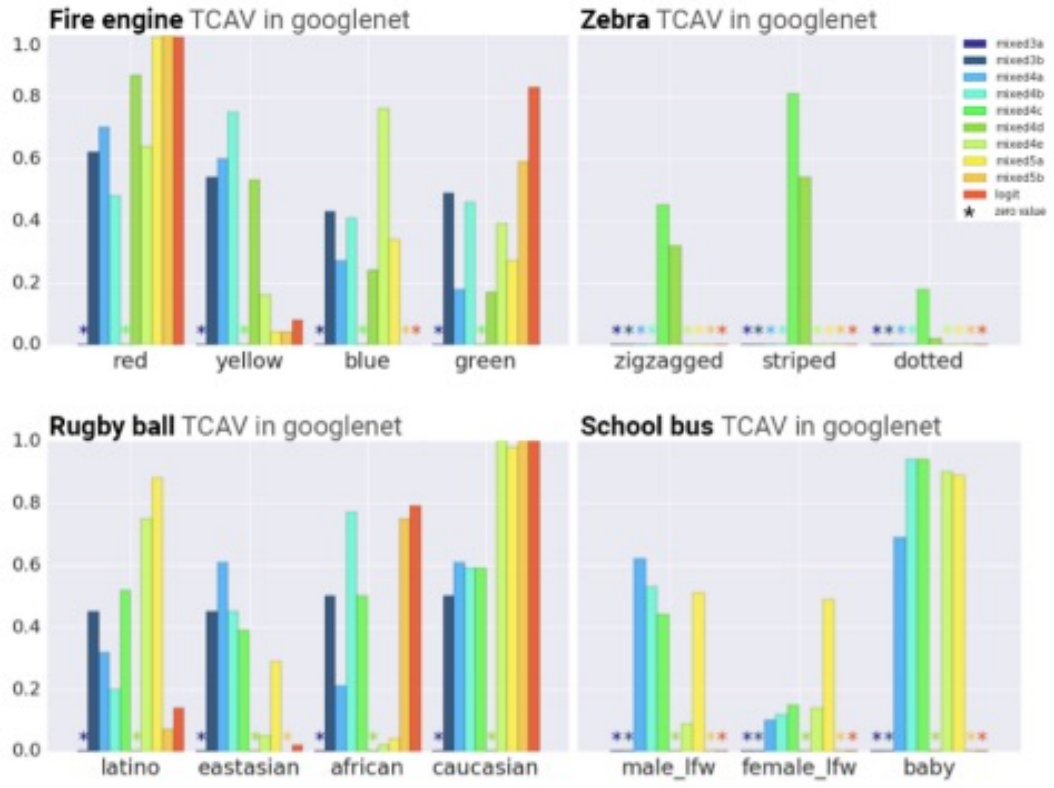
Portion of examples where concept C is positively related

X_k=



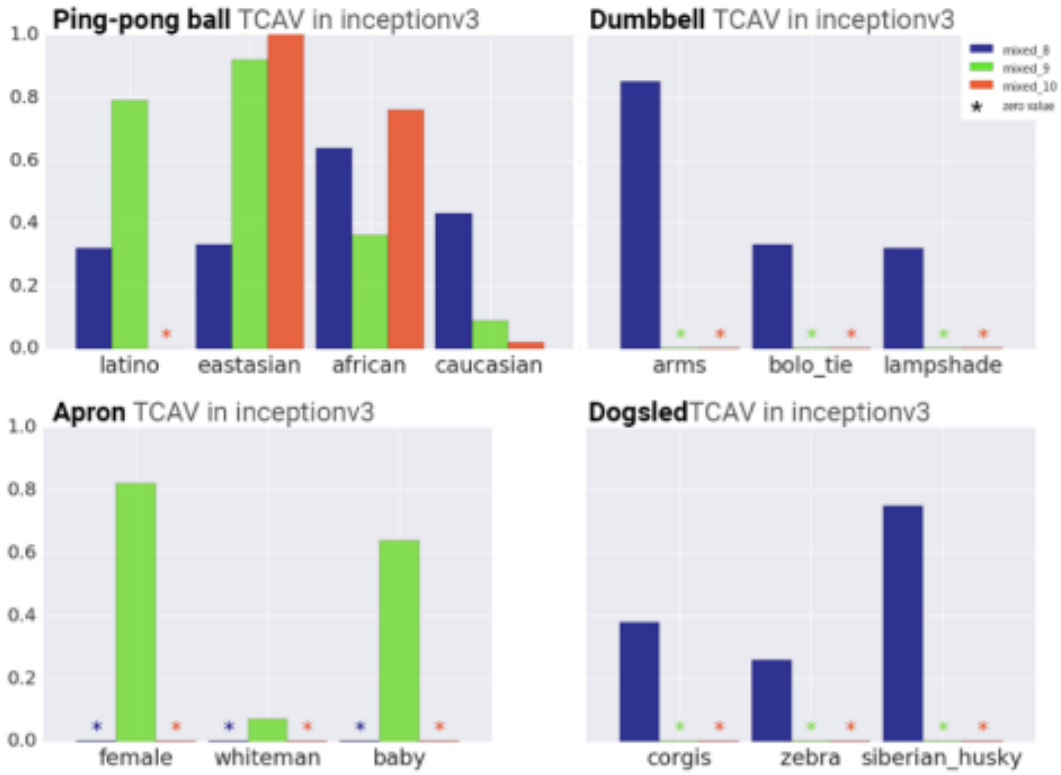
©2022 Su-In Lee

Example results



©2022 Su-In Lee

Example results



©2022 Su-In Lee

Remarks

- **Pros:**
  - TCAV is post-hoc, no architecture modifications
  - Fewer concept annotations required (but we still need examples to find CAVs)
- **Cons:**
  - Single direction (CAV) may not be able to represent complex concepts
  - Sensitivity to small changes may not be meaningful
  - Results depend on the layer

©2022 Su-In Lee

Today

- Section 1
  - Concept bottleneck models
  - Concept activation vectors
  - StylEx 
- Section 2
  - Neuron interpretation

©2022 Su-In Lee

Main idea

- Train a model that maps samples to *disentangled latent factors* (StyleGAN)
- Then, incorporate a classifier into the GAN
- Use humans to interpret each dimension of the StyleSpace as a concept (attribute)
- Generate attribute-wise counterfactuals, see how they impact the classifier

©2022 Su-In Lee

StyleGAN2

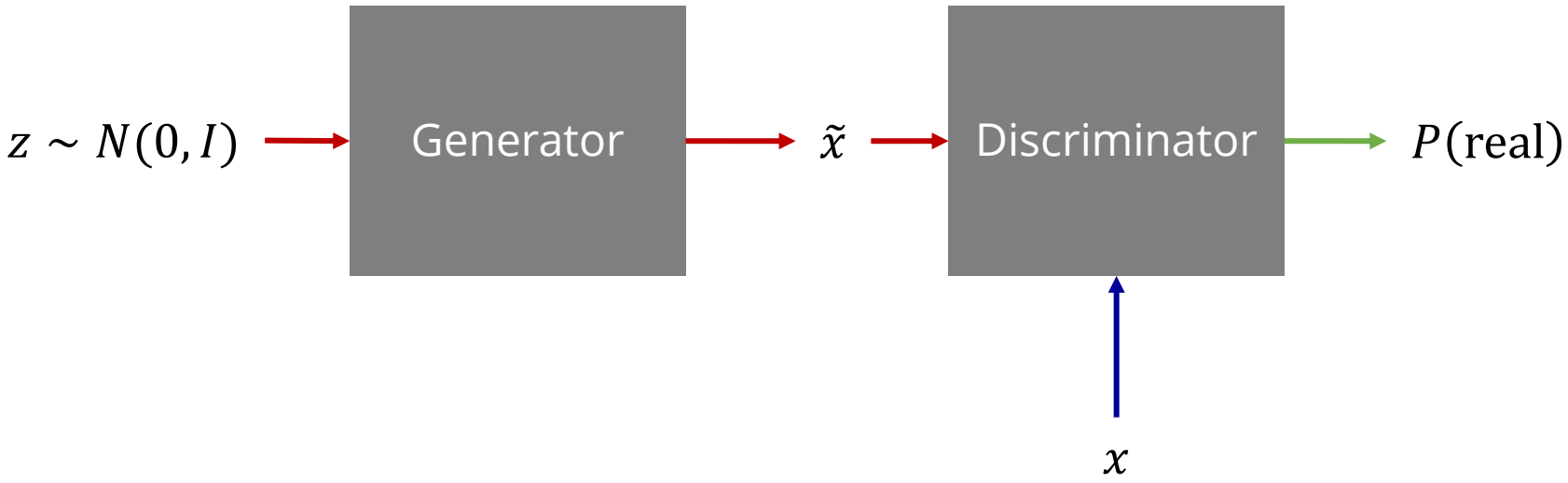
- A GAN architecture for generative image modeling, state-of-the-art performance in distribution quality metrics
- Produces a **disentangled latent space**
  - Latent dimensions correspond to high-level attributes (e.g., pose, freckles, hair)
  - Here, single dimensions rather than directions (like in TCAV)

Karras et al. "Analyzing and improving the image quality of StyleGAN" (2020)

©2022 Su-In Lee

StyleGAN2 (cont.)

- Basically, a GAN with improved architecture and training



Goodfellow et al. "Generative adversarial networks" (2014)

©2022 Su-In Lee

Example results

Fake people produced by StyleGAN2 generator

©2022 Su-In Lee

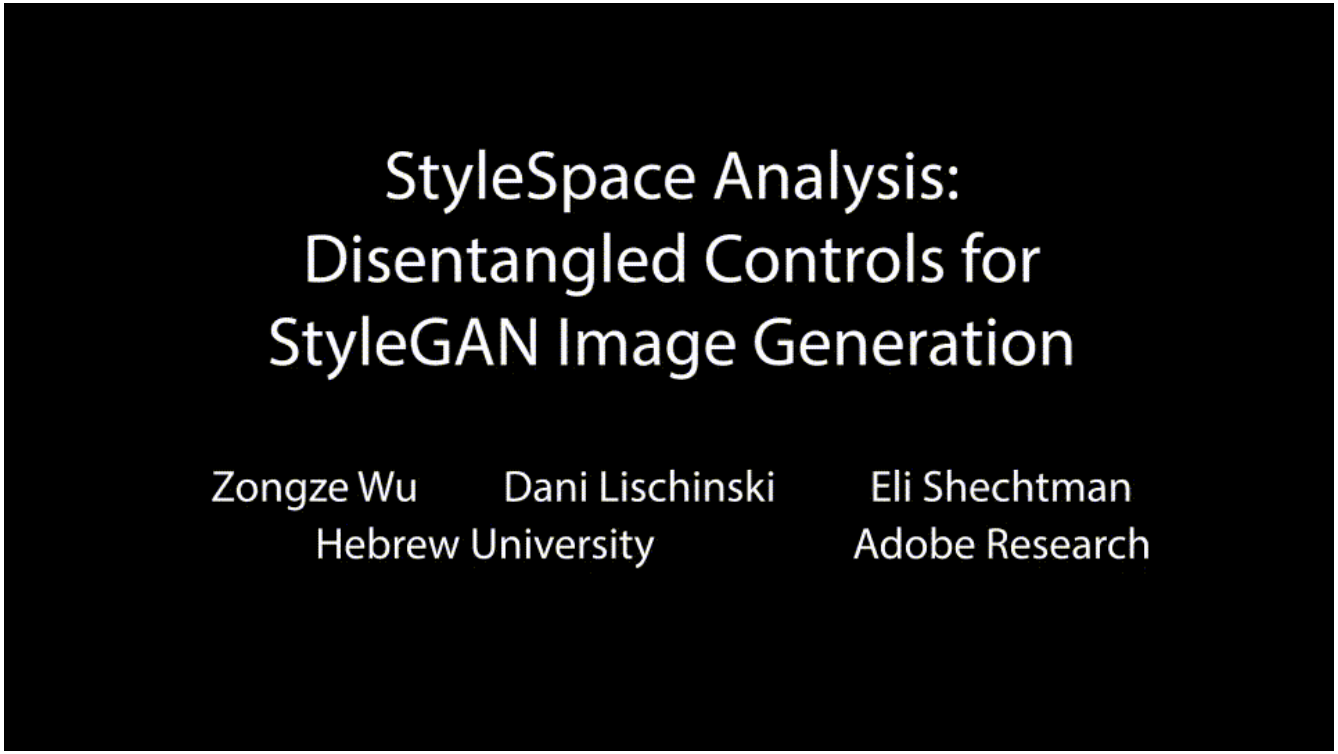
Observation: StyleSpace is disentangled

- Wu et al. explored an intermediate layer in StyleGAN2, called the “StyleSpace”
- Proposed using concept examples to identify dimensions that correspond to concepts (e.g., hair style, glasses)
- Then, adjusted these attributes to generate new images with desired properties

Wu et al., "StyleSpace analysis: Disentangled controls for StyleGAN image generation" (2021)

©2022 Su-In Lee

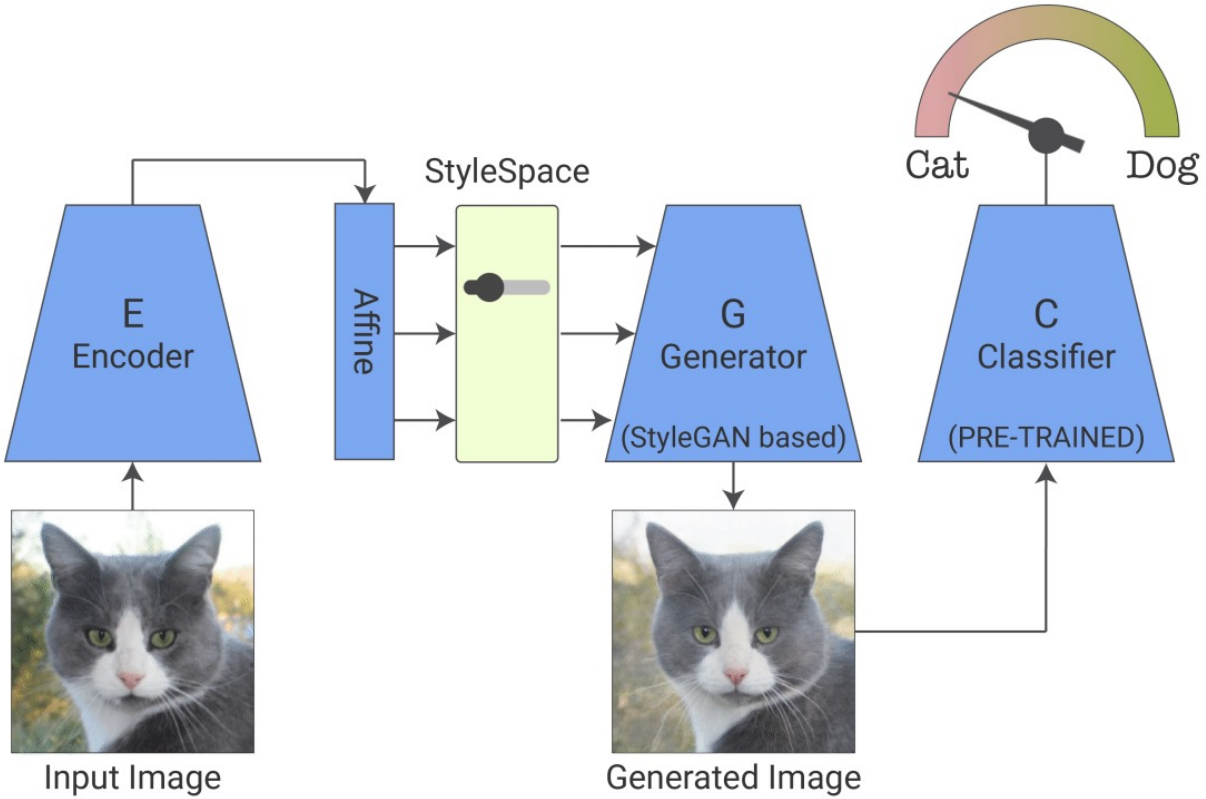
Observation: StyleSpace is disentangled



Wu et al., "StyleSpace analysis: Disentangled controls for StyleGAN image generation" (2021)

©2022 Su-In Lee

Latent space can represent concepts



Lang et al., "Explaining in style: Training a GAN to explain a classifier in StyleSpace" (2021)

©2022 Su-In Lee

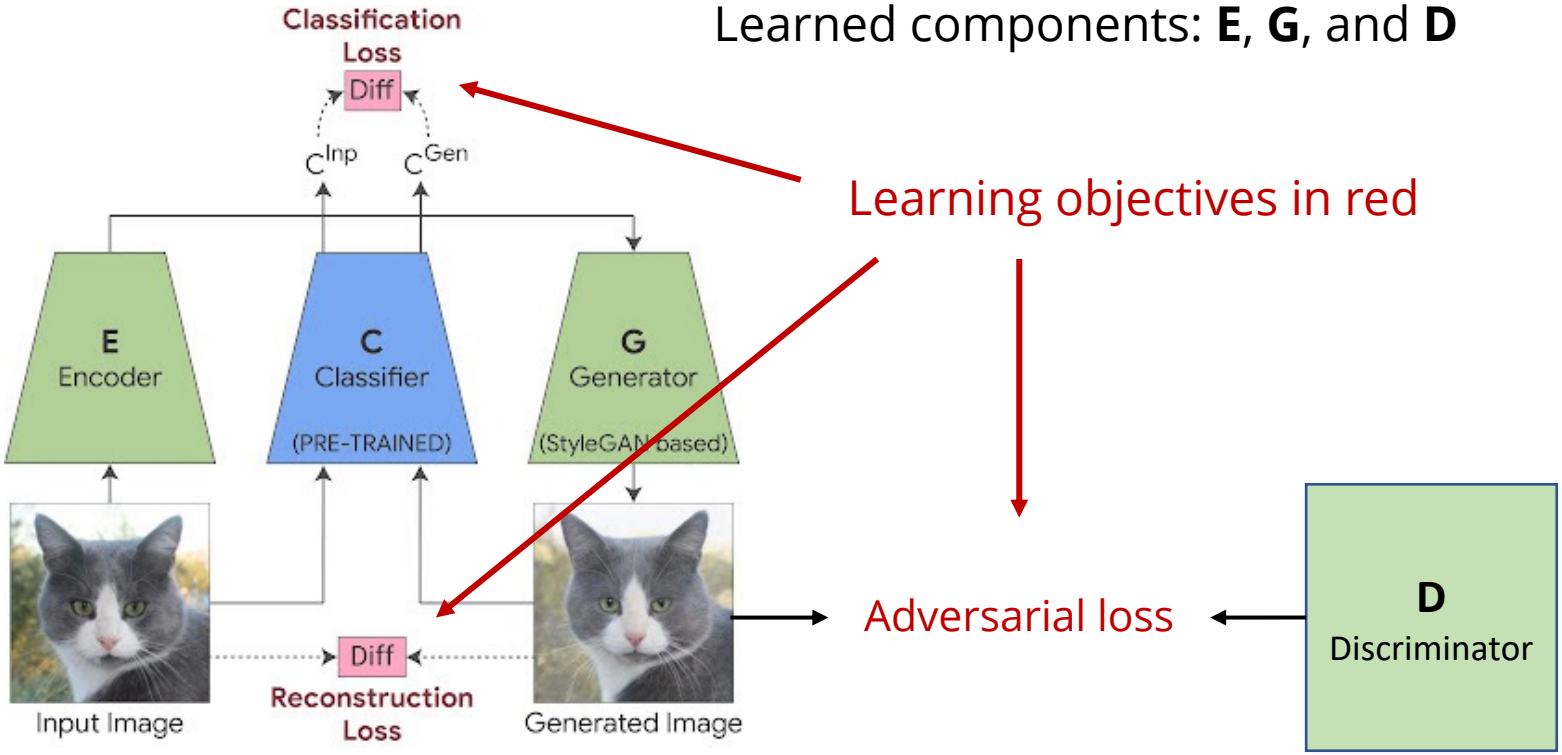
Combining classifier with StyleGAN2

- StyleGAN can produce attributes that don't affect the classifier
- StylEx proposed a StyleGAN training procedure that incorporates a classifier
  - Learns a classifier-specific StyleSpace
  - Classification loss ensures that generated image has same classification as corresponding original image

Lang et al., "Explaining in style: Training a GAN to explain a classifier in StyleSpace" (2021)

©2022 Su-In Lee

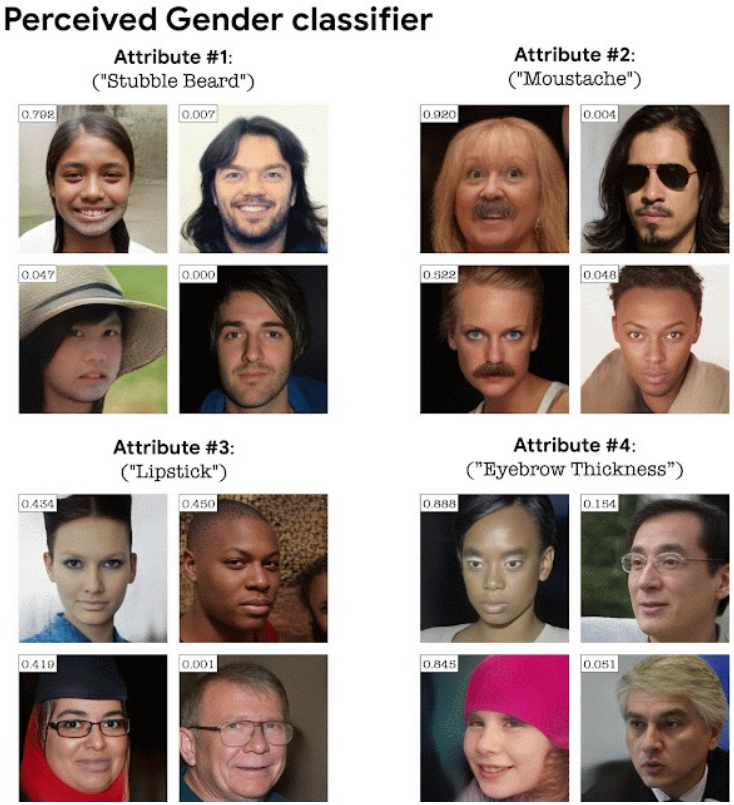
Combining classifier with StyleGAN2



Lang et al., "Explaining in style: Training a GAN to explain a classifier in StyleSpace" (2021)

©2022 Su-In Lee

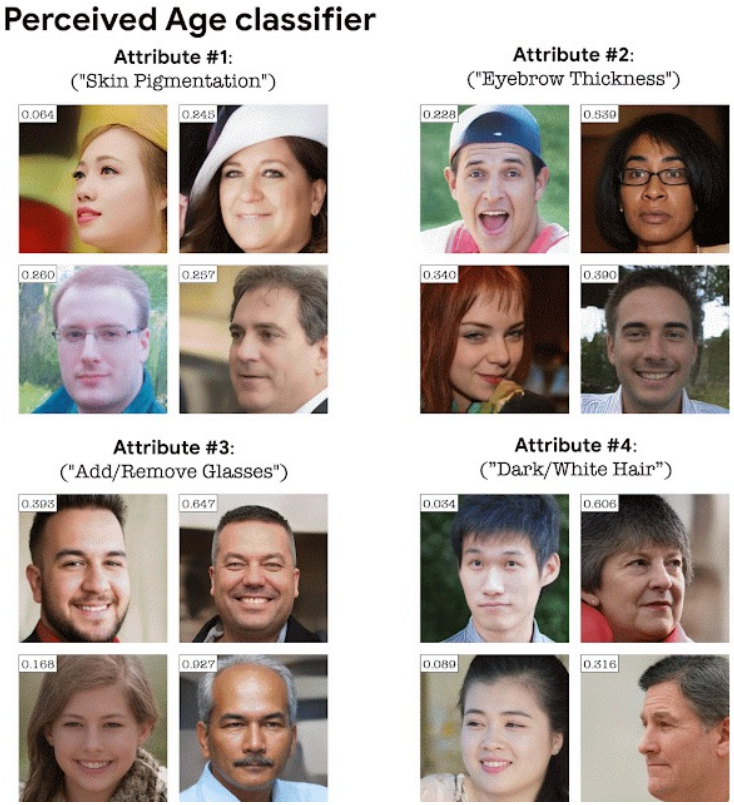
Example concepts in gender classifier



Lang et al., "Explaining in style: Training a GAN to explain a classifier in StyleSpace" (2021)

©2022 Su-In Lee

Example concepts in age classifier

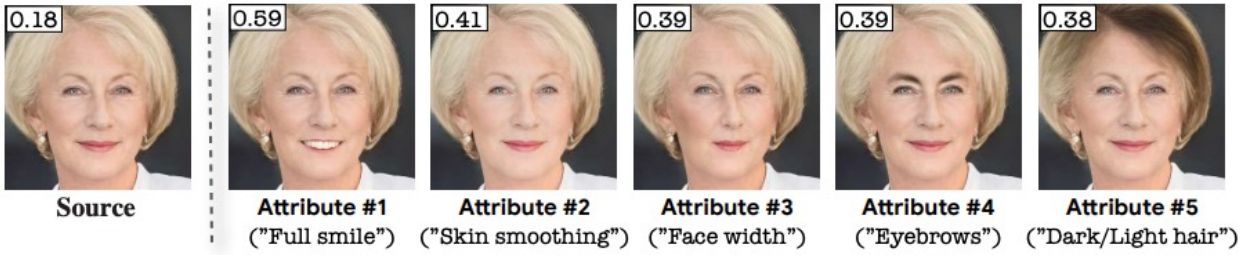


Lang et al., "Explaining in style: Training a GAN to explain a classifier in StyleSpace" (2021)

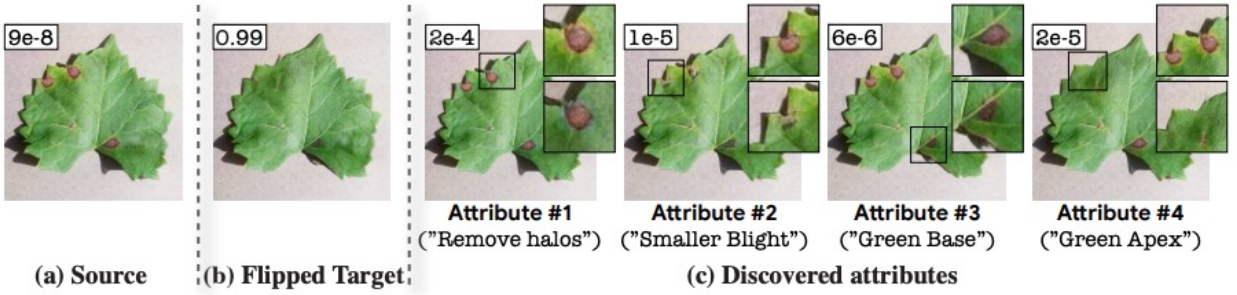
©2022 Su-In Lee

Local explanations

Independent



Interactions



Lang et al., "Explaining in style: Training a GAN to explain a classifier in StyleSpace" (2021)

©2022 Su-In Lee

Remarks

- **Pros:**
  - StyleGAN is trained without concept labels, concept directions are discovered automatically after training
- **Cons:**
  - GANs are difficult to train
  - Requires manual inspection to determine if latent space maps to disentangled factors (not guaranteed, works best for faces)
- Note: this can be considered a counterfactual explanation that changes one attribute at a time

©2022 Su-In Lee

Conclusion

- Concepts are not inherently explanations
- Concept explanations typically require two steps:
  - Learning a latent space of human understandable concepts
  - Explaining model predictions via that latent space

Approach	Concept annotation	Explanation	Learning approach
Concept bottleneck	All training samples	Intervention	Supervised
TCAV	Some samples	Directional derivative	Post-hoc supervised
StylEx	Some samples	Counterfactuals	Unsupervised

©2022 Su-In Lee

Today

- Section 1
  - Concept bottleneck models
  - Concept activation vectors
  - StylEx
  - **10 min break**
- Section 2
  - Neuron interpretation

©2022 Su-In Lee