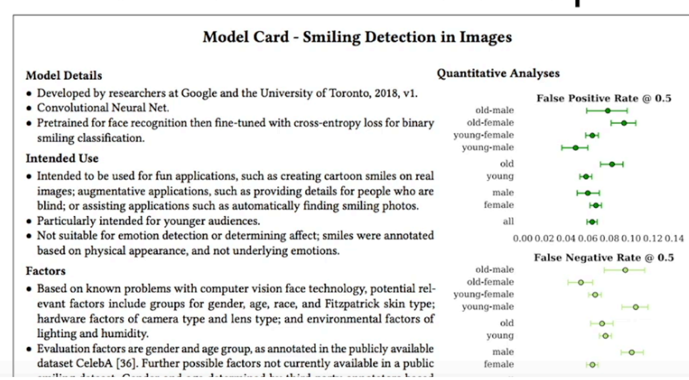
Tufts COMP 135: Introduction to Machine Learning

https://www.cs.tufts.educomp/135/2020f/

Responsible Machine Learning

Image credit: h2o.ai

Model Cards for Model Reporting



Model Card - Smiling Detection in Images

Model Details
- Developed by researchers at Google and the University of Toronto, 2018, v1.
- Convolutional Neural Net.
- Pretrained for face recognition then fine-tuned with cross-entropy loss for binary smiling classification.

Intended Use
- Intended to be used for fun applications, such as creating cartoon smiles on real images; augmentative applications, such as providing details for people who are blind; or assisting applications such as automatically finding smiling photos.
- Particularly intended for younger audiences.
- Not suitable for emotion detection or determining affect; smiles were annotated based on physical appearance, and not underlying emotions.

Factors
- Based on known problems with computer vision face technology, potential relevant factors include groups for gender, age, race, and Fitzpatrick skin type; hardware factors of camera type and lens type; and environmental factors of lighting and humidity.
- Evaluation factors are gender and age group, as annotated in the publicly available dataset CelebA [36]. Further possible factors not currently available in a public

Quantitative Analyses

Image credit: Mitchell et al. '19

Prof. Mike Hughes

Discussion Time!

See link to discussion guide on schedule

You have 15 minutes!

Responsible ML: Takeaways

- Human-centered approach
  - Involve diverse stakeholders at every step
- Cyclical approach
  - Plan to Test, Release, Revise, Repeat
- Know your data: make a Datasheet
  - What is it capable of?
  - Limitations?
- Know your model: make a Model Card
  - What is it capable of?
  - Limitations? (e.g. most ML can't make causal claims)

Review for Unit 5 Quiz

- SVMs for binary classification
  - Concepts: Support Vector, Hard Margin, Soft Margin
  - Hinge Loss
  - Compare/contrast with logistic regression

- Kernels
  - Definition of a kernel function
  - Examples: linear, squared-exponential, periodic
  - Primal vs dual view of prediction and training
  - Practical use: regression (like HW5), classification (with SVM)

- PCA
  - Encoding / decoding operations
  - Training objective
    - Minimize reconstruction error
  - Hyperparameter: How to select K?

- Fairness (at most 1 question)
  - How to tell if classifier is “fair”?
  - What metrics are appropriate?
    - Accuracy vs FPR vs TPR vs ….

Two views of kernel prediction

1. Pick a kernel **or** feature transform (one implies other)

$\phi(x_n) \in \mathbb{R}^G$

$$k(x_i, x_j) = \phi(x_i)^T \phi(x_j)$$

2. Then choose a view below (both not always possible)

Primal (weights view, explicit feature vectors)

$\theta \in \mathbb{R}^G$

$$\hat{y}(x_i, \theta) = \theta^T \phi(x_i) = \sum_{g=1}^{G} \theta_g \cdot \phi(x_i)_g$$

Dual (kernel view, only inner prod of features)

$\alpha \in \mathbb{R}^N$

$$\hat{y}(x_i, \alpha, \{x_n\}_{n=1}^N) = \sum_{n=1}^{N} \alpha_n k(x_n, x_i)$$

Why is kernel trick good idea?

Good with: squared exponential kernel
Not so good with: linear kernel with raw features

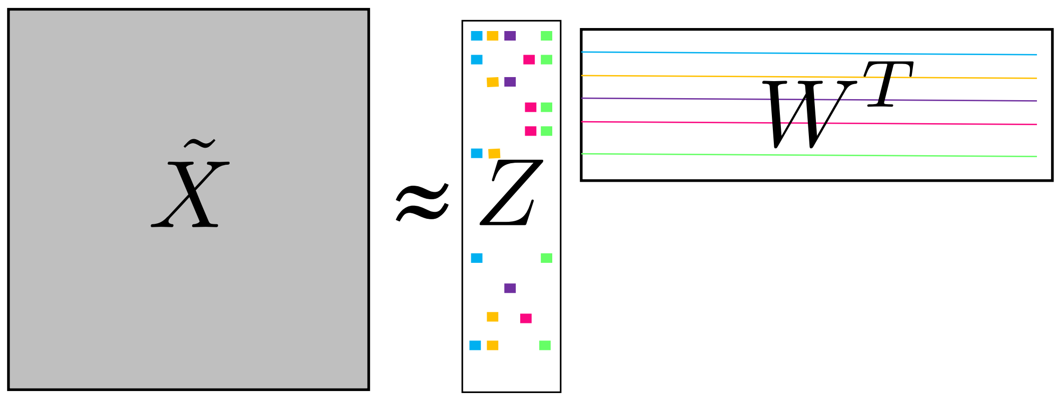
Before (primal view)
- Training problem seeks optimized vector of size G
- Prediction cost:
  - scales linearly with G (num. high-dim features)
  - requires G multiply ops plus G-1 adds

After (dual view)
- Training problem seeks optimized vector of size N
- Prediction cost:
  - scales linearly with N (num. train examples)
  - requires N evaluations of kernel plus N multiply/add

So we get some saving in runtime/storage with dual view if:
- G is bigger than N
- AND we can evaluate k fast (faster than a size G inner product)

Raw data: X Centered data: $\tilde{X}$ Reconstructed data: $\hat{X}$

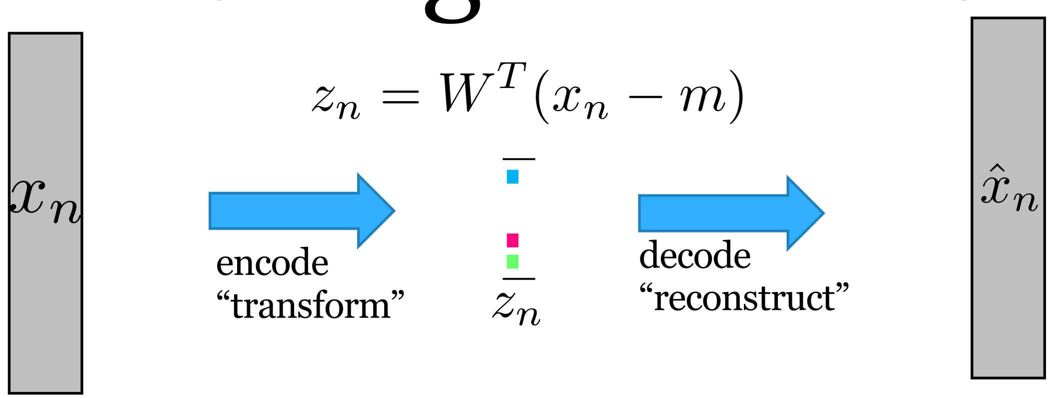
View: PCA as Matrix Factorization



$$\tilde{X} \approx Z W^T$$

$$W = \begin{bmatrix} | & | & & | \\ \mathbf{w}_1 & \mathbf{w}_2 & \dots & \mathbf{w}_K \\ | & | & & | \end{bmatrix}$$

View: Encoding and Decoding



$$z_n = W^T(x_n - m)$$

$$\hat{\mathbf{x}}_i = \mathbf{W}\mathbf{z}_i + \mathbf{m}$$

Breakout: Practice Quiz Unit 5