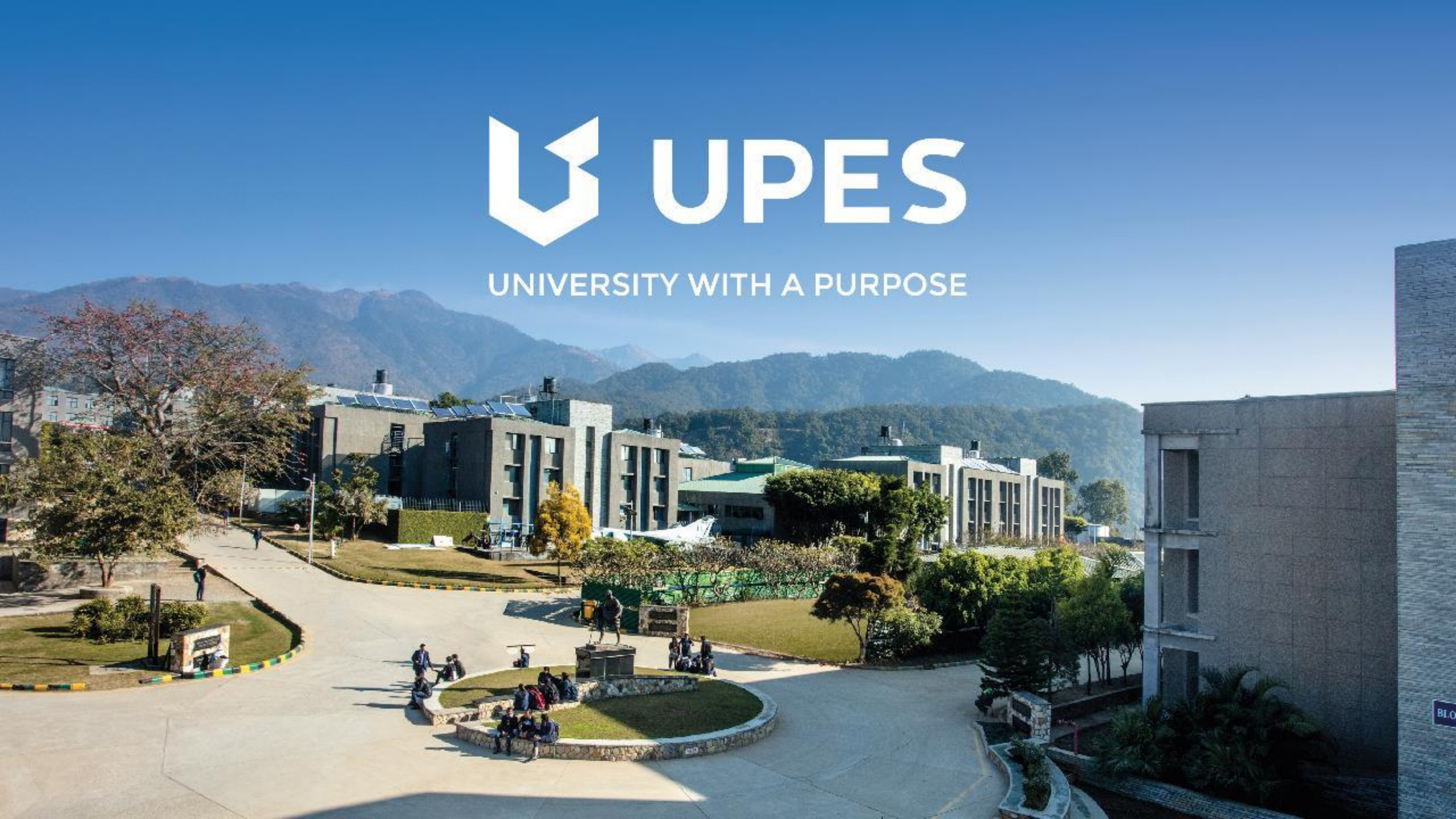
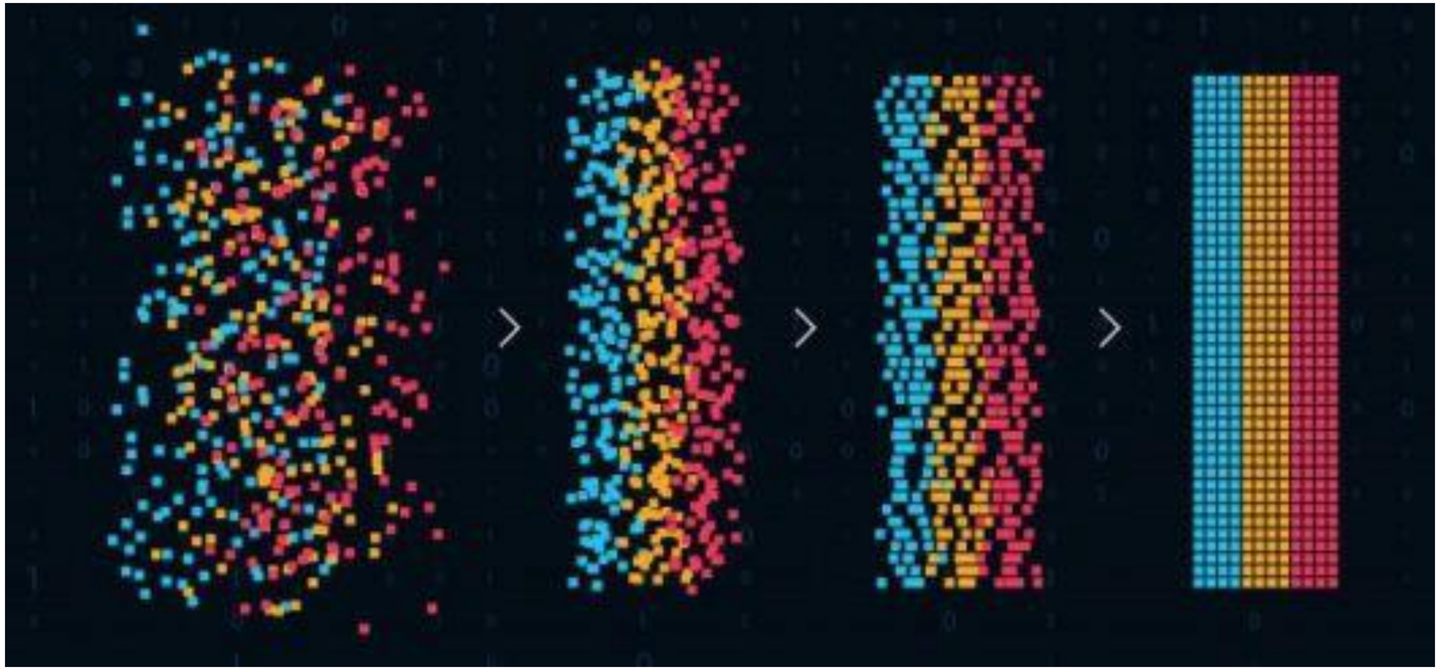

UPES
UNIVERSITY WITH A PURPOSE

Pattern and Anomaly Detection


Source: Edureka

B. Tech., CSE + AI/ML

Dr Gopal Singh Phartiyal

18/11/2021

Kernel Models

- So far: The linear and non-linear models developed do not use training data during prediction.
- Once the model is trained, the prediction is made on new input purely based on model parameters.
- Premise: How about involving training data during predictions.
- Parzen probability density model: linear combination of kernel functions cantered to each training data point.
- Nearest neighbours are another examples
- Generic termimology for such models: **Memory based**

Kernel Models

- Memory based methods: Store all training data to use it during prediction.
- These models always uses some measure or metric to indicate similarity between samples in input space.
- Fast to train but slow to predict

- A majority of linear models can be cast into equivalent "**Dual Representation**" where predictions are also based on linear combination of kernel functions evaluated at training samples.

Kernel Models

- For a fixed non-linear feature space, the kernel function is

$$k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^{\mathrm{T}}\phi(\mathbf{x}').$$

- If ø is 'linear', then

$$k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^{\mathrm{T}}\mathbf{x}'.$$

Is a linear kernel.

Imagine rebuilding the well-known models (NNs, PCA, SVMs, etc.) with this 'kernel' perspective. The approach is also known as kernel substitution.

Kernel Models

- Stationary kernels (not affected by translation in input space)

$$k(\mathbf{x}, \mathbf{x}') = k(\mathbf{x} - \mathbf{x}')$$

- Homogeneous kernels

$$k(\mathbf{x}, \mathbf{x}') = k(\|\mathbf{x} - \mathbf{x}'\|)$$

  - Example: radial basis function kernels

Kernel Models

- Ina a standard linear model, the regularized sum-of-squares error function is as follows

$$J(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\left\{\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n) - t_n\right\}^2 + \frac{\lambda}{2}\mathbf{w}^{\mathrm{T}}\mathbf{w}$$

- Solving for w yields

$$\mathbf{w} = -\frac{1}{\lambda}\sum_{n=1}^{N}\left\{\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n) - t_n\right\}\phi(\mathbf{x}_n)$$

$$= \sum_{n=1}^{N} a_n\phi(\mathbf{x}_n) = \mathbf{\Phi}^{\mathrm{T}}\mathbf{a}$$

- where

$$a_n = -\frac{1}{\lambda}\left\{\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n) - t_n\right\}$$

Kernel Models

- Writing the regularized sum-of-squares error function in terms of a.

$$J(\mathbf{a}) = \frac{1}{2}\mathbf{a}^{\mathrm{T}}\mathbf{\Phi\Phi}^{\mathrm{T}}\mathbf{\Phi\Phi}^{\mathrm{T}}\mathbf{a} - \mathbf{a}^{\mathrm{T}}\mathbf{\Phi\Phi}^{\mathrm{T}}\mathbf{t} + \frac{1}{2}\mathbf{t}^{\mathrm{T}}\mathbf{t} + \frac{\lambda}{2}\mathbf{a}^{\mathrm{T}}\mathbf{\Phi\Phi}^{\mathrm{T}}\mathbf{a}$$

- Define Gram matrix $\mathbf{K} = \mathbf{\Phi\Phi}^{\mathrm{T}}$

$$K_{nm} = \phi(\mathbf{x}_n)^{\mathrm{T}}\phi(\mathbf{x}_m) = k(\mathbf{x}_n, \mathbf{x}_m)$$

- Re-writing J(a) $J(\mathbf{a}) = \frac{1}{2}\mathbf{a}^{\mathrm{T}}\mathbf{KKa} - \mathbf{a}^{\mathrm{T}}\mathbf{Kt} + \frac{1}{2}\mathbf{t}^{\mathrm{T}}\mathbf{t} + \frac{\lambda}{2}\mathbf{a}^{\mathrm{T}}\mathbf{Ka}.$

$$\mathbf{a} = (\mathbf{K} + \lambda\mathbf{I}_N)^{-1}\mathbf{t}.$$

- Solving for a

$$\boxed{y(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}) = \mathbf{a}^{\mathrm{T}}\mathbf{\Phi}\phi(\mathbf{x}) = \mathbf{k}(\mathbf{x})^{\mathrm{T}}(\mathbf{K} + \lambda\mathbf{I}_N)^{-1}\mathbf{t}}$$

Kernel Models

- From here, you can make changes in any direction
  - Multiple output regression
  - Binary Classification
  - Multi-class classification
- Non-fixed basis function models
- Margin based models
- Projection based models

- Sparse kernel models

Next time: Combining Models for Pattern Recognition

Thank You

UPES
UNIVERSITY WITH A PURPOSE