Learning From Data
Lecture 10
Nonlinear Transforms

The $Z$-space
Polynomial transforms
Be careful

M. Magdon-Ismail
CSCI 4100/6100
RECAP: The Linear Model

linear in $\mathbf{x}$: gives the line/hyperplane separator

$$s = \mathbf{w}^T \mathbf{x}$$

linear in $\mathbf{w}$: makes the algorithms work
The Linear Model has its Limits

(a) Linear with outliers

(b) Essentially nonlinear

To address (b) we need something more than linear.
Change Your Features

\[ Y \gg 3 \text{ years} \]
no additional effect beyond \( Y = 3 \);

\[ Y \ll 0.3 \text{ years} \]
no additional effect below \( Y = 0.3 \).
Change Your Features Using a Transform

Years in Residence, $Y$

Income

Income

$z_1$

$Y$
Transform the data to a $\mathcal{Z}$-space in which the data is separable.

$$
\begin{align*}
\mathbf{x} &= \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix} \\
\mathbf{z} &= \Phi(\mathbf{x}) = \begin{bmatrix} 1 \\ x_1^2 \\ x_2^2 \end{bmatrix} = \begin{bmatrix} 1 \\ \Phi_1(\mathbf{x}) \\ \Phi_2(\mathbf{x}) \end{bmatrix}
\end{align*}
$$
Mechanics of the Feature Transform II

Separate the data in the $\mathcal{Z}$-space with $\tilde{w}$:

$$\tilde{g}(z) = \text{sign}(\tilde{w}^T z)$$
To classify a new $x$, first transform $x$ to $\Phi(x) \in \mathbb{Z}$-space and classify there with $\tilde{g}$.

$$g(x) = \tilde{g}(\Phi(x))$$
$$= \text{sign}(\tilde{w}^T\Phi(x))$$

$$\tilde{g}(z) = \text{sign}(\tilde{w}^Tz)$$
The General Feature Transform

$x$-space is $\mathbb{R}^d$

$\mathbf{x} = \begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_d \end{bmatrix}$

$x_1, x_2, \ldots, x_N$

$y_1, y_2, \ldots, y_N$

no weights

$g(\mathbf{x}) = \text{sign}(\mathbf{\tilde{w}}^T \Phi(\mathbf{x}))$

$\mathcal{Z}$-space is $\mathbb{R}^{\tilde{d}}$

$\mathbf{z} = \Phi(\mathbf{x}) = \begin{bmatrix} \Phi_1(\mathbf{x}) \\ \vdots \\ \Phi_{\tilde{d}}(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} 1 \\ z_1 \\ \vdots \\ z_{\tilde{d}} \end{bmatrix}$

$z_1, z_2, \ldots, z_N$

$y_1, y_2, \ldots, y_N$

$\mathbf{\tilde{w}} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_{\tilde{d}} \end{bmatrix}$
Generalization

\[ d_{VC} \quad \rightarrow \quad \tilde{d}_{VC} \]

\[ d + 1 \quad \rightarrow \quad \tilde{d} + 1 \]

Choose the feature transform with smallest \( \tilde{d} \)
Many Nonlinear Features May Work

\[ x_2 = x_1^2 + x_2^2 - 0.6 \]

\[ z_2 = x_2 \]

\[ z_1 = (x_1 + 0.05)^2 \]

\[ z_1 = x_1^2 \]

A rat! A rat!

This is called data snooping: looking at your data and tailoring your \( \mathcal{H} \).
Must Choose $\Phi$ **BEFORE** Your Look at the Data

After constructing features carefully, **before** seeing the data . . .

. . . if you think linear is not enough, try the 2nd order polynomial transform.

$$
\begin{bmatrix}
1 \\
x_1 \\
x_2
\end{bmatrix} = x \quad \rightarrow \quad \Phi(x) = \begin{bmatrix}
1 \\
\Phi_1(x) \\
\Phi_2(x) \\
\Phi_3(x) \\
\Phi_4(x) \\
\Phi_5(x)
\end{bmatrix} = \begin{bmatrix}
1 \\
x_1 \\
x_2 \\
x_1^2 \\
x_1x_2 \\
x_2^2
\end{bmatrix}
$$
The General Polynomial Transform $\Phi_k$

We can get even fancier: degree-$k$ polynomial transform:

$$\Phi_1(x) = (1, x_1, x_2),$$
$$\Phi_2(x) = (1, x_1, x_2, x_1^2, x_1 x_2, x_2^2),$$
$$\Phi_3(x) = (1, x_1, x_2, x_1^2, x_1 x_2, x_2^2, x_1^3, x_1^2 x_2, x_1 x_2^2, x_2^3),$$
$$\Phi_4(x) = (1, x_1, x_2, x_1^2, x_1 x_2, x_2^2, x_1^3, x_1^2 x_2, x_1 x_2^2, x_2^3, x_1^4, x_1^3 x_2, x_1 x_2^3, x_2^4),$$

\vdots

- Dimensionality of the feature space increases rapidly ($d_{vc}$)! 
- Similar transforms for $d$-dimensional original space. 
- Approximation-generalization tradeoff 
  
  Higher degree gives lower (even zero) $E_{in}$ but worse generalization.
Be Careful with Feature Transforms

Nonlinear Transforms
Be Careful with Feature Transforms

High order polynomial transform leads to "nonsense".
Digits Data “1” Versus “All”

Linear model
\[ E_{in} = 2.13\% \]
\[ E_{out} = 2.38\% \]

3rd order polynomial model
\[ E_{in} = 1.75\% \]
\[ E_{out} = 1.87\% \]
Use the Linear Model!

• First try a linear model – simple, robust and works.

• Algorithms can tolerate error plus you have nonlinear feature transforms.

• Choose a feature transform \textit{before} seeing the data. Stay simple. \textit{Data snooping is hazardous to your }E_{\text{out}}. \textit{.}

• Linear models are fundamental in their own right; they are also the building blocks of many more complex models like support vector machines.

• Nonlinear transforms also apply to regression and logistic regression.