IML Tutorial 2

Vignesh Ram Somnath
vsomnath@ethz.ch
Overview

- Fitting non-linear functions
- Feature selection
- Min-norm solution
- Computation of population risk
Fitting non-linear functions

So far we looked at functions linear in both $w$ and $x$

Linearity in this case comes from $w$

So far, we looked at functions of the form

$$y = w^T x + \varepsilon$$

$$y = ax^2 + bx + c$$

$$= [a \ b \ c] \begin{bmatrix} x^2 \\ x \\ 1 \end{bmatrix}$$

$$= \hat{w}^T x$$ (Now non-linear in $x$)
Fitting non-linear functions

Can fit non-linear functions via linear regression, by using non-linear features of our data

\[ f(x) = \sum_{i=1}^{d} w_i \phi_i(x) \]

\[ x \in \mathbb{R}^d \]

\[ x \mapsto \phi(x) \in \mathbb{R}^p \]

Consider the case: \( d = 1, p = 3 \)

\[ \phi_1(x) = (1, x, x^2, x^3) \]

\[ X = \begin{bmatrix}
\phi_1(x) \\
\phi_2(x) \\
\vdots \\
\phi_n(x)
\end{bmatrix} = \begin{bmatrix}
1, x_1, x_1^2, x_1^3 \\
1, x_2, x_2^2, x_2^3 \\
\vdots \\
1, x_n, x_n^2, x_n^3
\end{bmatrix} \]

We can use the same techniques as linear regression that we learnt
Fitting non-linear functions

Check provided code and play around with it
Fitting non-linear functions

Takeaways:

• Extend linear (in $w$) regression to fitting non-linear (in $x$) models

• Demonstrated this for function approximation using polynomial features

Code will be uploaded, feel free to tinker with it!
Feature Selection
Feature Selection

We may not want to work with all available features

Preference for a subset of features:

- **Interpretability** - Identifying important variables / features
- **Generalization** - Simpler models may generalize better
Feature Selection

Filter Methods
Identify relevant features by estimating some distributional quantity (no model training)

e.g. Choose all features whose variance is above a certain threshold

Wrapper Methods
Identify “useful” features based on model performance (usually cross-validation)

e.g. Select a subset of features, evaluate model performance by cross validation

Intrinsic / Embedded Methods
Identify useful features jointly with model learning / construction

e.g. Lasso - $L_1$ penalty encourages sparsity
Feature Selection

Greedy Feature Selection

Basic Idea:
Greedily add (or remove) features to improve some scoring function

Setting:
Consider a set $V = \{1, 2, \ldots, d\}$ - Denotes feature indices

Scoring function $CV(S)$ (cross validation error for subset $S$)
Greedy Forward Selection

Given: $S = \phi$ and $E_0 = \infty$ ($E_0$ is the initial error)

For $i = 1: d$

Find best element to add
\[ s_i = \text{argmin}_{j \in V \setminus S} L(S \cup \{j\}) \]

Compute error
\[ E_i = L(S \cup \{j\}) \]

if $E_i > E_{i-1}$, break
else $S \leftarrow S \cup \{s_i\}$
Greedy Backward Selection

Given: $S = V$ and $E_{d+1} = \infty$

For $i = d:-1:1$

Find best element to remove
$s_i = \arg\min_{j \in S} L(S \setminus \{j\})$

Compute error
$E_i = L(S \setminus \{j\})$

if $E_i > E_{i+1}$, break
else $S \leftarrow S \setminus \{s_i\}$
# Greedy Feature Selection

<table>
<thead>
<tr>
<th>Forward Selection</th>
<th>Backward Selection</th>
</tr>
</thead>
<tbody>
<tr>
<td>Typically fast if fewer relevant features</td>
<td>Better at handling feature dependencies</td>
</tr>
<tr>
<td>Dependency handling between features is poor</td>
<td>Need to train several models for higher dimensions</td>
</tr>
</tbody>
</table>

Greedy solutions may be **suboptimal**

*Check out accompanying code!*
Derivation of the minimum-norm solution

Given \( X \in \mathbb{R}^{n \times d} \), \( w \in \mathbb{R}^d \) and \( y \in \mathbb{R}^n \), our objective function takes the form:

\[
\hat{w} = \arg \min_{w \in \mathbb{R}^d} ||y - Xw||_2^2
\]

Consider the case \( n < d \). Such a system of equations are called underdetermined.

**Claim:** When \( n < d \), \( Xw = y \) has infinitely many solutions.

From rank-nullity theorem,

\[
\text{rank}(X) + \text{Nullity}(X) = d
\]

\[
\text{Nullity}(X) = d - \text{rank}(X) > d - n > 0
\]

\[
\Rightarrow \forall v \in \mathbb{R}^d \text{ s.t. } Xv = 0.
\]

Consider a particular solution \( w_p \in \mathbb{R}^d \) of \( Xw = y \)

\[
w' = w_p + cv \quad \forall c \in \mathbb{R}
\]

is also a solution of \( Xw = y \)

\[
Xw' = X(w + cv) = Xw + cXv = Xw + 0 = Xw
\]

\[
\Rightarrow \text{For } n < d, \ Xw = y \text{ has infinitely many solutions}
\]

We need an additional constraint to choose a solution from the available set of solutions.
A commonly used constraint is that of min-norm.

\[
\begin{align*}
\text{min} \quad \|w\| & \quad \text{s.t.} \quad Xw = y
\end{align*}
\]

**CLAIM:** The min-norm solution \( W_{\text{min}} \in \text{span}(X^T) \).

Before proving the above claim, we show that \( \text{span}(X^T) \) and \( \text{Null}(X) \) are orthogonal subspaces.

**PROOF:**

\[
\forall v \in \text{Null}(X) \Rightarrow Xv = 0
\]

Consider an arbitrary vector \( z \in \mathbb{R}^n \).

\[
\langle z, Xv \rangle = z^T(Xv)
\]

\[
= z^T(X^T)^Tv
\]

\[
= (X^Tz)^T v
\]

As \( Xv = 0 \), \( \langle z, Xv \rangle = 0 \).

\[
\Rightarrow (X^Tz)^T v = 0
\]

\( X^Tz \) consists of all vectors \( m \in \text{span}(X^T) \).

Hence \( \forall m \in \text{span}(X^T) \), \( \langle m, v \rangle = 0 \).

\[
\Rightarrow \text{span}(X^T) \text{ and } \text{Null}(X) \text{ are orthogonal to each other}
\]

Using the above fact, we can decompose \( \mathbb{R}^d \) into two subspaces that are orthogonal.
Consider a vector \( w_p \) s.t. \( Xw_p = y \). This vector has two components \( w_{\text{null}} \in \text{Null}(X) \) and \( w_{\text{span}(X^T)} \) The norm can be decomposed into those of its components as spaces are orthogonal \( \|
abla p\|_2^2 = \|w_{\text{null}}\|_2^2 + \|w_{\text{span}(X^T)}\|_2^2 \) From (1) and (2), we see that the minimum norm solution occurs only when \( w_{\text{null}} = 0 \), i.e. there is no projection of \( w_p \) onto \( \text{Null}(X) \). This is only possible when \( w_p \in \text{span}(X^T) \)

\[
\Rightarrow \hat{w}_m \in \text{span}(X^T) \Rightarrow \forall \, v \in \mathbb{R}^n \, \text{s.t.} \, \hat{w}_m = X^T v
\]

\[
X \hat{w}_m = y
\]
\[
X X^T v = y
\]

\[
\Rightarrow v = (X X^T)^+ y \quad \text{where} \quad (A^+)^T \text{denotes pseudo-inverse}
\]
\[ \Rightarrow \hat{\omega}_{mn} = X^T v = (X^T x)^+ x^T \]

**Bonus Claim:** \[ x^T (x x^T)^+ = (x^T x)^+ x^T \]

**Proof:**

By using SVD decomposition,

\[ X = U \Sigma V^T \quad \text{where} \quad U \in \mathbb{R}^{n \times n}, \ \Sigma \in \mathbb{R}^{n \times d}, \ V \in \mathbb{R}^{d \times d} \quad \text{and} \quad U^T U = I, \ V^T V = I \]

\[ a) \] \[ x x^T = (U \Sigma V^T) (U \Sigma V^T)^T \]

\[ = U \Sigma U^T (V^T)^T \Sigma^T V^T \]

\[ = U \Sigma V^T \Sigma^T V^T \]

\[ = U \Sigma \Sigma^T U^T \]

\[ (x x^T)^+ = (U \Sigma \Sigma^T U^T)^+ \]

\[ = (U^T)^+ (\Sigma \Sigma^T)^+ U^T \]

\[ = U (\Sigma \Sigma^T)^+ U^T \quad \quad \quad \quad \quad \text{(As} \ V^T U = I, U \Sigma U^T \text{are inverses of each other)} \]

\[ x^T (x x^T)^+ = \sum^T U^T U (\Sigma \Sigma^T)^+ U^T \]

\[ = \sum^T (\Sigma \Sigma^T)^+ U^T \]

\[ b) \] \[ x^T x = \sum^T U^T U \sum \sum V^T = \sum^T \sum V^T \]

\[ (x^T x)^+ = (V^T)^+ (\Sigma^T \Sigma)^+ V^+ = V (\Sigma^T \Sigma)^+ V^T \]
\[(X^T X)^+ X^T = V (\Sigma^T \Sigma)^+ V^T \Sigma^T U^T\]

Consider the expressions marked within the text. Because \( \Sigma \) only has elements along the diagonal, \((\Sigma^T \Sigma)^+ \Sigma^T\) and \(\Sigma^T (\Sigma \Sigma^T)^+\) result in the same final expressions.

\[
\Sigma \in \mathbb{R}^{n \times d}
\]

\[
\Sigma^T \Sigma \in \mathbb{R}^{d \times d}
\]

\[
(\Sigma^T \Sigma)^+ \in \mathbb{R}^{d \times d}
\]

\[
(\Sigma^T \Sigma)^+ \Sigma^T \in \mathbb{R}^{d \times n}
\]

One can show similarly for \( \Sigma^T (\Sigma \Sigma^T)^+ \).

When \( d < n \), we saw

\[
\hat{\omega} = (X^T X)^+ X^T y
\]
Now, we got the min-norm solution.
\[ \hat{w}_{mn} = X^T(XX^T)^+ y \]

The last part showed that
\[ X^T(XX^T)^+ = (X^T y)^T X^T \]

\[ \Rightarrow \hat{w}_{mn} = (X^T X)^+ X^T y. \]

With this, we can write
\[ \hat{w}_{LS} = (X^T X)^+ X^T y \quad \forall \ n,d \in \mathbb{Z}^+ \]
Derivation of the population risk

Given a function \( f : \mathbb{R}^d \rightarrow \mathbb{R} \), \( x \in \mathbb{R}^d \) and \( y \in \mathbb{R} \)

\[
R(f) = \int \int \ell(f(x), y) \ p(x, y) \ dx \ dy
\]

where \( \ell : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R} \) is an arbitrary loss function

\[
R(f) = E_{(x,y) \sim p(x,y)} \left[ \ell(f(x), y) \right]
\]

Setting

let \( x \in \mathbb{R}^d \) and \( x \sim N(0, I_d) \).

Let \( y = x^T \hat{\omega} + \tilde{\epsilon} \) where \( \tilde{\epsilon} \sim N(0, \sigma^2) \) and \( \hat{\omega} \in \mathbb{R}^d \) are the true weights.

For the training set \( y = X\omega^* + \epsilon \) where \( \epsilon \sim N(0, \sigma^2 I_d) \)

**NOTE**: \( \epsilon \) and \( \tilde{\epsilon} \) are i.i.d.

let the inputs \( X \) be fixed. Our goal is to then compute \( E_{\epsilon}[R(f)] \) where \( f : \mathbb{R}^d \rightarrow \mathbb{R} \) and \( f(x) = x^T \hat{\omega} \)

where,

\[
\hat{\omega} = (X^TX)^+ X^T y
\]

\[
\Rightarrow \hat{\omega} = (X^TX)^+ X^T (X\omega^* + \epsilon)
\]

\[
\Rightarrow \hat{\omega} = (X^TX)^+ X^T \omega^* + (X^TX)^+ X^T \epsilon
\]

\[ (1) \]
\[ E_\varepsilon[RC(\hat{f})] = E_{\varepsilon, x, y} \left[ \| y - x^T \hat{\omega} \|^2 \right] \]
\[ = E_{\varepsilon, x, \varepsilon} \left[ \| x^T \hat{\omega} + \varepsilon - x^T \hat{\omega} \|^2 \right] \]
\[ = E_{\varepsilon, x, \varepsilon} \left[ \| x^T (\omega^* - \hat{\omega}) \|^2 \right] + E_{\varepsilon, x, \varepsilon} [\varepsilon^2] \]
\[ - 2E_{\varepsilon, x, \varepsilon} \left[ x^T (\omega^* - \hat{\omega}) \cdot \varepsilon \right] \]

Consider the cross term \( 2E_{\varepsilon, x, \varepsilon} \left[ x^T (\omega^* - \hat{\omega}) \cdot \varepsilon \right] \)
As \( \varepsilon \perp x \perp \varepsilon \),
\[ 2E_{\varepsilon, x, \varepsilon} \left[ x^T (\omega^* - \hat{\omega}) \cdot \varepsilon \right] = 2E_{\varepsilon, x} \left[ x^T (\omega^* - \hat{\omega}) E_{\varepsilon} [\varepsilon] \right] \]
\[ = 0 \]

\[ \Rightarrow E_\varepsilon[RC(\hat{f})] = E_{\varepsilon, x, \varepsilon} \left[ \| x^T (\omega^* - \hat{\omega}) \|^2 \right] + E_{\varepsilon} [\varepsilon^2] \]

The expression inside the expectation is not dependent on \( \varepsilon \).

\[ = E_{\varepsilon, x} \left[ \| x^T (\omega^* - \hat{\omega}) \|^2 \right] + E_{\varepsilon} [\varepsilon^2] \]
\[ = E_{\varepsilon} \left[ (\omega^* - \hat{\omega})^T E_x [xx^T] (\omega^* - \hat{\omega}) \right] + \sigma^2 \]

\[ I \text{ by definition} \]
\[
\begin{align*}
&= E_\varepsilon [\|w^* - \hat{w}\|^2] + \sigma^2 \\
\text{Using (1), we get} &
= E_\varepsilon [\|w^* - Tw^* - K\varepsilon\|^2] + \sigma^2 \\
&
= E_\varepsilon [\|w^* - Tw^*\|^2] + E_\varepsilon [\|K\varepsilon\|^2] + \sigma^2
\end{align*}
\]

(The cross term is eliminated using similar arguments as before)

\[
\begin{align*}
&= \|w^* - Tw^*\|^2 + E_\varepsilon [\varepsilon^T X (X^T X)^+ (X^T X)^+ X^T \varepsilon] + \sigma^2
\end{align*}
\]

Consider \( E_\varepsilon [\varepsilon^T X (X^T X)^+ (X^T X)^+ X^T \varepsilon] \)

\textit{Scalar}

As the term inside the expectation is a scalar, we can apply the cyclicity of trace to get

\[
\begin{align*}
&= E_\varepsilon [\text{tr} (\varepsilon \varepsilon^T X (X^T X)^+ (X^T X)^+ X^T)] \\
&= \text{tr} (E_\varepsilon (\varepsilon \varepsilon^T) X (X^T X)^+ (X^T X)^+ X^T) \\
&= \sigma^2 \text{tr} (X (X^T X)^+ (X^T X)^+ X^T) \\
&= \sigma^2 \text{tr} ((X^T X)^+ (X^T X) (X^T X)^+) \\
&= \sigma^2 \text{tr} ((X^T X)^+)\tag{Apply cyclicity twice}
\end{align*}
\]

By definition of pseudo inverse
\( A^+ A A^+ = A^+ \)

\[
\begin{align*}
&= \sigma^2 \text{tr} ((X^T X)^+)
\end{align*}
\]
By SVD (see the min-norm example above)

\[ X^T X = \Sigma^T \Sigma \Sigma^T \]

\[ \Rightarrow \sigma^2 \text{tr}((X^T X)^+) = \sigma^2 \text{tr}(\Sigma^T \Sigma)^+ \text{tr}(\Sigma^T \Sigma) \]

\[ = \sigma^2 \text{tr}(\text{tr} \Sigma^T \Sigma)^+ \]

\[ = \sigma^2 \text{tr}(\Sigma^T \Sigma)^+ \]

\[ = \sum_{i: \sigma_i \neq 0} \sigma_i^2 / \sigma_i^2 \]

Now, consider \( \| w^* - Tw^* \|^2 \) where \( T = (X^T X)^+(X^T X) \).

Recall from lecture 2 that

\[ \Pi_X = X(X^T X)^+ X^T \quad \forall \text{ any } X \]

\[ \Rightarrow \Pi_{X^T} = X^T (XX^T)^+ X \]

From a previous result for min-norm (BONUS CLAIM), we showed,

\[ X^T (XX^T)^+ = (X^T X)^+ X^T \]

Multiplying by \( X \) on the right, we get

\[ X^T (XX^T)^+ X = (X^T X)^+ X^T X \]

\[ \Rightarrow \Pi_{X^T} = T \]

Our final expression thus looks like,
$$E_{\varepsilon[R(\hat f)]]} = \| w^* - \Pi_{X^T} w^* \|^2 + \sum_{i: \sigma_i \neq 0} \frac{\sigma_i^2}{\sigma_i^2} + \sigma_i^2$$

Consider $T_1$: $T_1$ is large when $\Pi_{X^T} w^*$ is small

$\Rightarrow \dim(\text{span}(X^T)) = \dim(\text{span}(X))$ is small

$\Rightarrow X$ has smaller rank.

Consider $T_2$: $T_2$ is large when $\frac{1}{\sigma_i^2}$ is small,

$\Rightarrow$ some directions are not as well represented in $\text{span}(X)$

END OF TUTORIAL 😊