Introduction to Machine Learning

Regularization, Gradient Descent and Fisher Linear Discriminant

Andres Mendez-Vazquez

May 23, 2019

Outline

Cinvestav

Outline

Cinvestav

Well-Posed Problem

Definition by Hadamard (Circa 1902)

- Models of physical phenomenas should have the following properties
  1. A solution exists,
  2. The solution is unique,
  3. The solution's behavior changes continuously with the initial conditions.

Any other problem that fails in any of this conditions

- It is considered an Ill-Posed Problem.

Well-Posed Problem

Definition by Hadamard (Circa 1902)

- Models of physical phenomenas should have the following properties
  1. A solution exists,
  2. The solution is unique,
  3. The solution's behavior changes continuously with the initial conditions.

Any other problem that fails in any of this conditions

- It is considered an Ill-Posed Problem.

Regularization in Linear Problems

In many applications of linear algebra

We want to find and estimation $\widehat{\boldsymbol{x}}$ to a vector $\boldsymbol{x} \in \mathbb{R}^d$ satisfying the approximation

$$Ax \approx y$$

When $A \in \mathbb{R}^{m \times d}$ is ill-conditioned or singular.

Regularization in Linear Problems

In many applications of linear algebra

We want to find and estimation $\widehat{\boldsymbol{x}}$ to a vector $\boldsymbol{x} \in \mathbb{R}^d$ satisfying the approximation

$$A\boldsymbol{x} \approx \boldsymbol{y}$$

When $A \in \mathbb{R}^{m \times d}$ is ill-conditioned or singular.

The importance of the problem

The problems generating these situations are:

1. Numerical differentiation of noisy data,
2. Non parametric smoothing of curves and surfaces defined by scattered data,
3. Image reconstruction,
4. Inverse Laplace transforms,
5. etc.

Regularization in Linear Problems

In many applications of linear algebra

We want to find and estimation $\widehat{\boldsymbol{x}}$ to a vector $\boldsymbol{x} \in \mathbb{R}^d$ satisfying the approximation

$$A\boldsymbol{x} \approx \boldsymbol{y}$$

When $A \in \mathbb{R}^{m \times d}$ is ill-conditioned or singular.

The importance of the problem

The problems generating these situations are:

1. Numerical differentiation of noisy data,
2. Non parametric smoothing of curves and surfaces defined by scattered data,
3. Image reconstruction,
4. Inverse Laplace transforms,
5. etc.

Regularization in Linear Problems

In many applications of linear algebra

We want to find and estimation $\widehat{\boldsymbol{x}}$ to a vector $\boldsymbol{x} \in \mathbb{R}^d$ satisfying the approximation

$$A\boldsymbol{x} \approx \boldsymbol{y}$$

When $A \in \mathbb{R}^{m \times d}$ is ill-conditioned or singular.

The importance of the problem

The problems generating these situations are:

1. Numerical differentiation of noisy data,
2. Non parametric smoothing of curves and surfaces defined by scattered data,
3. Image reconstruction,
4. Inverse Laplace transforms,
5. etc.

Regularization in Linear Problems

In many applications of linear algebra

We want to find and estimation $\widehat{\boldsymbol{x}}$ to a vector $\boldsymbol{x} \in \mathbb{R}^d$ satisfying the approximation

$$A\boldsymbol{x} \approx \boldsymbol{y}$$

When $A \in \mathbb{R}^{m \times d}$ is ill-conditioned or singular.

The importance of the problem

The problems generating these situations are:

1. Numerical differentiation of noisy data,
2. Non parametric smoothing of curves and surfaces defined by scattered data,
3. Image reconstruction,
4. Inverse Laplace transforms,
5. etc.

Regularization in Linear Problems

In many applications of linear algebra

We want to find and estimation $\widehat{\boldsymbol{x}}$ to a vector $\boldsymbol{x} \in \mathbb{R}^d$ satisfying the approximation

$$A\boldsymbol{x} \approx \boldsymbol{y}$$

When $A \in \mathbb{R}^{m \times d}$ is ill-conditioned or singular.

The importance of the problem

The problems generating these situations are:

1. Numerical differentiation of noisy data,
2. Non parametric smoothing of curves and surfaces defined by scattered data,
3. Image reconstruction,
4. Inverse Laplace transforms,
5. etc.

Regularization in Linear Problems

In many applications of linear algebra

We want to find and estimation $\widehat{\boldsymbol{x}}$ to a vector $\boldsymbol{x} \in \mathbb{R}^d$ satisfying the approximation

$$A\boldsymbol{x} \approx \boldsymbol{y}$$

When $A \in \mathbb{R}^{m \times d}$ is ill-conditioned or singular.

The importance of the problem

The problems generating these situations are:

1. Numerical differentiation of noisy data,
2. Non parametric smoothing of curves and surfaces defined by scattered data,
3. Image reconstruction,
4. Inverse Laplace transforms,
5. etc.

Regularization in Linear Problems

In many applications of linear algebra

We want to find and estimation $\widehat{\boldsymbol{x}}$ to a vector $\boldsymbol{x} \in \mathbb{R}^d$ satisfying the approximation

$$A\boldsymbol{x} \approx \boldsymbol{y}$$

When $A \in \mathbb{R}^{m \times d}$ is ill-conditioned or singular.

The importance of the problem

The problems generating these situations are:

1. Numerical differentiation of noisy data,
2. Non parametric smoothing of curves and surfaces defined by scattered data,
3. Image reconstruction,
4. Inverse Laplace transforms,
5. etc.

Regularization in Linear Problems

In many applications of linear algebra

We want to find and estimation $\widehat{\boldsymbol{x}}$ to a vector $\boldsymbol{x} \in \mathbb{R}^d$ satisfying the approximation

$$A\boldsymbol{x} \approx \boldsymbol{y}$$

When $A \in \mathbb{R}^{m \times d}$ is ill-conditioned or singular.

The importance of the problem

The problems generating these situations are:

1. Numerical differentiation of noisy data,
2. Non parametric smoothing of curves and surfaces defined by scattered data,
3. Image reconstruction,
4. Inverse Laplace transforms,
5. etc.

In all such situations

The Vector $\widehat{x}$ generated by

1. $\widehat{\boldsymbol{x}} = A^{-1}\boldsymbol{y}$
2. $\widehat{\boldsymbol{x}} = \left(A^{T}A\right)^{-1}A^{T}\boldsymbol{y}$

If it exists at all

- It is usually a meaningless bad approximation to x.

In all such situations

The Vector $\widehat{\boldsymbol{x}}$ generated by

1. $\widehat{\boldsymbol{x}} = A^{-1}\boldsymbol{y}$
2. $\widehat{\boldsymbol{x}} = \left(A^T A\right)^{-1} A^T \boldsymbol{y}$

If it exists at all

- It is usually a meaningless bad approximation to $\boldsymbol{x}$.

Even

Even with an estimation $\widehat{\boldsymbol{x}} = A\boldsymbol{y}$ as reasonable near to $\boldsymbol{x}^*$ (Square Case)

$$
\begin{aligned}
\|\boldsymbol{x}^* - \widehat{\boldsymbol{x}}\| &= \left\|A^{-1}A\boldsymbol{x}^* - A^{-1}\boldsymbol{y}\right\| \\
&\leq \left\|A^{-1}\right\| \|A\boldsymbol{x}^* - \boldsymbol{y}\| \text{ Holder's Inequality}
\end{aligned}
$$

- This Upper Bound is quite large.

Even

Even with an estimation $\widehat{\boldsymbol{x}} = A\boldsymbol{y}$ as reasonable near to $\boldsymbol{x}^*$ (Square Case)

$$
\begin{aligned}
\left\|\boldsymbol{x}^* - \widehat{\boldsymbol{x}}\right\| &= \left\|A^{-1}A\boldsymbol{x}^* - A^{-1}\boldsymbol{y}\right\| \\
&\leq \left\|A^{-1}\right\| \left\|A\boldsymbol{x}^* - \boldsymbol{y}\right\| \text{ Holder's Inequality}
\end{aligned}
$$

- This Upper Bound is quite large.

With

1. $\left\|A^{-1}\right\| = \sigma_{\max}(A)$ The largest singular value of matrix.
2. $\left\|A\boldsymbol{x} - \boldsymbol{y}\right\| = \sqrt{(A\boldsymbol{x} - \boldsymbol{y})^T (A\boldsymbol{x} - \boldsymbol{y})}$

Even

Even with an estimation $\widehat{\boldsymbol{x}} = A\boldsymbol{y}$ as reasonable near to $\boldsymbol{x}^*$ (Square Case)

$$
\begin{aligned}
\left\|\boldsymbol{x}^* - \widehat{\boldsymbol{x}}\right\| &= \left\|A^{-1}A\boldsymbol{x}^* - A^{-1}\boldsymbol{y}\right\| \\
&\leq \left\|A^{-1}\right\| \left\|A\boldsymbol{x}^* - \boldsymbol{y}\right\| \text{ Holder's Inequality}
\end{aligned}
$$

- This Upper Bound is quite large.

With

1. $\left\|A^{-1}\right\| = \sigma_{\max}(A)$ The largest singular value of matrix.
2. $\left\|A\boldsymbol{x} - \boldsymbol{y}\right\| = \sqrt{(A\boldsymbol{x} - \boldsymbol{y})^T (A\boldsymbol{x} - \boldsymbol{y})}$

Even

Even with an estimation $\widehat{\boldsymbol{x}} = A\boldsymbol{y}$ as reasonable near to $\boldsymbol{x}^*$ (Square Case)

$$\begin{aligned}\left\|\boldsymbol{x}^* - \widehat{\boldsymbol{x}}\right\| &= \left\|A^{-1}A\boldsymbol{x}^* - A^{-1}\boldsymbol{y}\right\| \\ &\leq \left\|A^{-1}\right\| \left\|A\boldsymbol{x}^* - \boldsymbol{y}\right\| \text{ Holder's Inequality}\end{aligned}$$

- This Upper Bound is quite large.

With

1. $\left\|A^{-1}\right\| = \sigma_{\max}(A)$ The largest singular value of matrix.
2. $\left\|A\boldsymbol{x} - \boldsymbol{y}\right\| = \sqrt{(A\boldsymbol{x} - \boldsymbol{y})^T (A\boldsymbol{x} - \boldsymbol{y})}$

Even

Even with an estimation $\widehat{\boldsymbol{x}} = A\boldsymbol{y}$ as reasonable near to $\boldsymbol{x}^*$ (Square Case)

$$
\begin{aligned}
\left\|\boldsymbol{x}^* - \widehat{\boldsymbol{x}}\right\| &= \left\|A^{-1}A\boldsymbol{x}^* - A^{-1}\boldsymbol{y}\right\| \\
&\leq \left\|A^{-1}\right\| \left\|A\boldsymbol{x}^* - \boldsymbol{y}\right\| \text{ Holder's Inequality}
\end{aligned}
$$

- This Upper Bound is quite large.

With

1. $\left\|A^{-1}\right\| = \sigma_{\max}(A)$ The largest singular value of matrix.
2. $\left\|A\boldsymbol{x} - \boldsymbol{y}\right\| = \sqrt{(A\boldsymbol{x} - \boldsymbol{y})^T (A\boldsymbol{x} - \boldsymbol{y})}$

Therefore

Regularization techniques are needed to obtain meaningful solutions

- To problems that are called ill-posed problems.

Where some parameters are ill-determined

- By Least Square Methods
  - in particular when the number of parameters is larger than the number of available measurements!!!

Cinvestav

Therefore

Regularization techniques are needed to obtain meaningful solutions

- To problems that are called ill-posed problems.

Where some parameters are ill-determined

- By Least Square Methods
  - in particular when the number of parameters is larger than the number of available measurements!!!

Outline

Modeling Smoothness

Geometrically, regularization for smoothness means that

- We seek the least rough function that gives a certain degree of fit to the observed data.

A way to measure smoothness

- It is look at how many derivatives can be done before $\nabla^p f(x) = 0$

Modeling Smoothness

Geometrically, regularization for smoothness means that

- We seek the least rough function that gives a certain degree of fit to the observed data.

A way to measure smoothness

- It is look at how many derivatives can be done before $\nabla^p f(x) = 0$

Cinvestav

Here, we want to model the idea of “Smoothness”

For this, we consider a continuous function f

- Where we use a vector $\boldsymbol{w}$ with features

$$w_i = f\left(t_i\right)$$

Thus, we can use a numerical differentiation method such that

$$w^{(1)} = \frac{df\left(t\right)}{dt}$$

Here, we want to model the idea of "Smoothness"

For this, we consider a continuous function f

- Where we use a vector $\boldsymbol{w}$ with features

$$w_i = f(t_i)$$

Thus, we can use a numerical differentiation method such that

$$\boldsymbol{w}^{(1)} = \frac{df(t)}{dt}$$

Cinvestav

Therefore, Assume Smoothness

We have a value such that $w = f(t)$

- Thus, we say that w is smooth "enough" if $w^{(1)} = \frac{df(t)}{dt}$ exists.

Now this can be repeated p

$$w^{(p)} = \frac{d^{(p)} f(t)}{dt^{(p)}}$$

Therefore, Assume Smoothness

We have a value such that $w = f(t)$

- Thus, we say that w is smooth "enough" if $w^{(1)} = \frac{df(t)}{dt}$ exists.

Now this can be repeated p

$$w^{(p)} = \frac{d^{(p)} f(t)}{dt^{(p)}}$$

Thus, it is possible to look at this smoothness

Using our Linear Algebra, we can represent this as a Linear Operator

$$w^{(p)} = Sw \text{ (The Smoothing Matrix)}$$

Thus

We can define the numerical differentiation of a $p+1$ times

- Over a continuously differentiable function

$$y:[0,1]\longrightarrow \mathbb{R}$$

Thus, finding our estimate $x(t)=y'(t)=\nabla y(t)$

- Basically our problem of solving the linear system $Ax=y$

Or in other words

$$Ax(t)=\int_0^t x(\tau)\,d\tau$$

Thus

We can define the numerical differentiation of a $p+1$ times

- Over a continuously differentiable function

$$y : [0,1] \longrightarrow \mathbb{R}$$

Thus, finding our estimate $x(t) = y'(t) = \nabla y(t)$

- Basically our problem of solving the linear system $Ax = y$

Or in other words

$$Ax(t) = \int_0^t x(\tau)\, d\tau$$

Cinvestav

Thus

We can define the numerical differentiation of a $p+1$ times

- Over a continuously differentiable function

$$y : [0,1] \longrightarrow \mathbb{R}$$

Thus, finding our estimate $x(t) = y'(t) = \nabla y(t)$

- Basically our problem of solving the linear system $Ax = y$

Or in other words

$$Ax(t) = \int_0^t x(\tau)\, d\tau$$

Cinvestav

Therefore

The differentiability assumption says

$$\boldsymbol{w} = \nabla^{p+1} y = \nabla^{p} x \text{ is continous and bounded}$$

Given that $A = \nabla^{-1}$

- We may write the previous equation as

$$x = A^{p} w$$

Therefore

The differentiability assumption says

$$\boldsymbol{w} = \nabla^{p+1} y = \nabla^{p} x \text{ is continous and bounded}$$

Given that $A = \nabla^{-1}$

- We may write the previous equation as

$$x = A^{p}\boldsymbol{w}$$

Furthermore, Based in the following equalities

We can define the Adjoint Integral Operator is defined $\left\langle A^T x_1, x_2 \right\rangle = \langle x_1, Ax_2 \rangle$

$$\langle x_1, x_2 \rangle = \int_0^1 x_1(t)\, x_2(t)\, dt$$

Thus with $Ax_i = y_i$ and $x_i = \nabla y_i$

$$\langle x_1, Ax_2 \rangle = \langle \nabla y_1, y_2 \rangle = -\langle y_1, \nabla y_2 \rangle = \langle -Ax_1, x_2 \rangle$$

- By Partial Integration

Then, under the following boundary conditions

- Assuming that y and its first $p+1$ derivatives vanish at $t=0$ and $t=1$.

Furthermore, Based in the following equalities

We can define the Adjoint Integral Operator is defined $\left\langle A^T x_1, x_2 \right\rangle = \langle x_1, Ax_2 \rangle$

$$\langle x_1, x_2 \rangle = \int_0^1 x_1(t)\, x_2(t)\, dt$$

Thus with $Ax_i = y_i$ and $x_i = \nabla y_i$

$$\langle x_1, Ax_2 \rangle = \langle \nabla y_1, y_2 \rangle = -\langle y_1, \nabla y_2 \rangle = \langle -Ax_1, x_2 \rangle$$

- By Partial Integration

Then, under the following boundary conditions

- Assuming that y and its first $p+1$ derivatives vanish at $t=0$ and $t=1$.

Furthermore, Based in the following equalities

We can define the Adjoint Integral Operator is defined $\left\langle A^T x_1, x_2 \right\rangle = \langle x_1, A x_2 \rangle$

$$\langle x_1, x_2 \rangle = \int_0^1 x_1(t)\, x_2(t)\, dt$$

Thus with $A x_i = y_i$ and $x_i = \nabla y_i$

$$\langle x_1, A x_2 \rangle = \langle \nabla y_1, y_2 \rangle = -\langle y_1, \nabla y_2 \rangle = \langle -A x_1, x_2 \rangle$$

- By Partial Integration

Then, under the following boundary conditions

- Assuming that y and its first $p+1$ derivatives vanish at $t = 0$ and $t = 1$.

How?

We have

$$\langle \nabla y_1, y_2 \rangle = \int_0^1 \nabla y_1(t)\, y_2(t)\, dt$$

$$= y_1(t)\, y_2(t)\big|_0^1 - \int_0^1 y_1(t)\, \nabla y_2(t)\, dt$$

$$= -\langle y_1, \nabla y_2 \rangle$$

Cinvestav

How?

We have

$$\langle \nabla y_1, y_2 \rangle = \int_0^1 \nabla y_1(t)\, y_2(t)\, dt$$

$$= y_1(t)\, y_2(t)\, |_0^1 - \int_0^1 y_1(t)\, \nabla y_2(t)\, dt$$

$$= -\langle y_1, \nabla y_2 \rangle$$

Then, if we assume that all entries in A are in $\mathbb{R}$

- $A^T = -A$

How?

We have

$$\begin{aligned}\langle \nabla y_1, y_2 \rangle &= \int_0^1 \nabla y_1(t)\, y_2(t)\, dt \\ &= y_1(t)\, y_2(t)\, |_0^1 - \int_0^1 y_1(t)\, \nabla y_2(t)\, dt \\ &= -\langle y_1, \nabla y_2 \rangle\end{aligned}$$

Then, if we assume that all entries in A are in $\mathbb{R}$

- $A^T = -A$

Cinvestav

Therefore

We have the following relation

$$\nabla y(t) = A^{-1} y(t)$$

Thus, it is possible to write the condition $x = A^p w$ as $x = Sw$

- By absorbing the sign into w

$$S = \begin{cases} \left(A^T A\right)^{\frac{p}{2}} & \text{if } p \text{ is even} \\ \left(A^T A\right)^{\frac{p-1}{2}} A^T & \text{if } p \text{ is even} \end{cases}$$

- For $p \geq 1$.

Therefore

We have the following relation

$$\nabla y(t) = A^{-1} y(t)$$

Thus, it is possible to write the condition $x = A^p w$ as $x = Sw$

- By absorbing the sign into w

$$S = \begin{cases} \left(A^T A\right)^{\frac{p}{2}} & \text{if } p \text{ is even} \\ \left(A^T A\right)^{\frac{p-1}{2}} A^T & \text{if } p \text{ is even} \end{cases}$$

- For $p \geq 1$.

Outline

Cinvestav

The key to the treatment of ill-posed Linear Systems

It is a process called regularization that replaces A^{-1} by a family $C_h, h > 0$

- Of approximate inverses of A in such a way that, as $h \longrightarrow 0$, the product $C_h A \to I$ in an appropriately restricted sense.
  - The parameter h is called the regularization parameter.

Therefore

It is usually possible to choose the C_h such that

- For a suitable exponent p (often $p = 1$ or 2), the constants
  1. $\gamma_1 = \sup_{h>0} h \left\| C_h \right\|$.
  2. $\gamma_2 = \sup_{h>0} h^{-p} \left\| \left(I - C_h A \right) S \right\|$

They are finite and of reasonable size

- From this... we have...

Therefore

It is usually possible to choose the C_h such that

- For a suitable exponent p (often $p = 1$ or 2), the constants
  1. $\gamma_1 = \sup_{h>0} h \|C_h\|$.
  2. $\gamma_2 = \sup_{h>0} h^{-p} \|(I - C_h A) S\|$

They are finite and of reasonable size

- From this... we have...

Cinvestav

The Following Theorem

Theorem

- Suppose $x = Sw$, and $\|Ax - y\| \leq \Delta \|w\|$ for some $\Delta > 0$.
- Then γ_1 and γ_2 implies

$$\|x - C_h y\| \leq \left[\gamma_1 \frac{\Delta}{h} + \gamma_2 h^p\right] \|w\|$$

Cinvestav

The Following Theorem

Theorem

- Suppose $x = Sw$, and $\|Ax - y\| \leq \Delta \|w\|$ for some $\Delta > 0$.
- Then γ_1 and γ_2 implies

$$\|x - C_h y\| \leq \left[\gamma_1 \frac{\Delta}{h} + \gamma_2 h^p\right] \|w\|$$

The Following Theorem

Theorem

- Suppose $x = Sw$, and $\|Ax - y\| \leq \Delta \|w\|$ for some $\Delta > 0$.
- Then γ_1 and γ_2 implies

$$\|x - C_h y\| \leq \left[\gamma_1 \frac{\Delta}{h} + \gamma_2 h^p\right] \|w\|$$

Cinvestav

For Example

For a well-posed data fitting problem

- One with a well-conditioned normal equation matrix A^TA

The least squares estimate

- It has an error of the order of Δ.

For example

- $C_h = (A^TA)^{-1}A^T = A^+ \Longrightarrow h^{-1} = \|A^+\| = O(1)$ with $\gamma_1 = 1$ and $\gamma_2 = 0$ independent of p

For Example

For a well-posed data fitting problem

- One with a well-conditioned normal equation matrix $A^T A$

The least squares estimate

- It has an error of the order of Δ.

For example

- $C_h = \left(A^T A\right)^{-1} A^T = A^+ \Longrightarrow h^{-1} = \|A^+\| = O(1)$ with $\gamma_1 = 1$ and $\gamma_2 = 0$ independent of p

For Example

For a well-posed data fitting problem

- One with a well-conditioned normal equation matrix $A^T A$

The least squares estimate

- It has an error of the order of Δ.

For example

- $C_h = (A^T A)^{-1} A^T = A^+ \Longrightarrow h^{-1} = \|A^+\| = O(1)$ with $\gamma_1 = 1$ and $\gamma_2 = 0$ independent of p

Cinvestav

Outline

Cinvestav

Now, we have

When A is rank deficient or becomes increasingly ill-conditioned

- We may improve the condition by modifying $A^T A$.

The simplest way to achieve this is by adding a small multiple of the identity

- Since $A^T A$ is symmetric and positive semidefinite.

The matrix $A^T A + h^2 I$ has its eigenvalues

- They are in the interval $\left[h^2, h^2 + \|A\|^2\right]$

Now, we have

When A is rank deficient or becomes increasingly ill-conditioned

- We may improve the condition by modifying $A^T A$.

The simplest way to achieve this is by adding a small multiple of the identity

- Since $A^T A$ is symmetric and positive semidefinite.

The matrix $A^T A + h^2 I$ has its eigenvalues

- They are in the interval $\left[h^2, h^2 + \|A\|^2\right]$

Now, we have

When A is rank deficient or becomes increasingly ill-conditioned

- We may improve the condition by modifying $A^T A$.

The simplest way to achieve this is by adding a small multiple of the identity

- Since $A^T A$ is symmetric and positive semidefinite.

The matrix $A^T A + h^2 I$ has its eigenvalues

- They are in the interval $\left[h^2, h^2 + \|A\|^2\right]$

Cinvestav

Here

The Condition Number of a Positive Definite Matrix Σ

$$cond\left(\Sigma\right) = \frac{\lambda_{max}\left(\Sigma\right)}{\lambda_{min}\left(\Sigma\right)}$$

- What happens

Which is related to the Maximum Likelihood of a Gaussian Distribution under a restriction

$$\max ML\left(\Sigma\right)$$
$$s.t. cond\left(\Sigma\right) \leq k$$

- "Condition Number Regularized Covariance Estimation" by Won et.. al

Here

The Condition Number of a Positive Definite Matrix Σ

$$cond\left(\Sigma\right)=\frac{\lambda_{max}\left(\Sigma\right)}{\lambda_{min}\left(\Sigma\right)}$$

- What happens

Which is related to the Maximum Likelihood of a Gaussian Distribution under a restriction

$$\begin{aligned}&\max\ ML\left(\Sigma\right)\\&s.t. cond\left(\Sigma\right)\leq k\end{aligned}$$

- "Condition Number Regularized Covariance Estimation" by Won et. al

Here the Condition Number

It is

$$cond\left(A^T A + h^2 I\right) \leq \frac{h^2 + \|A\|^2}{h^2}$$

Therefore, we have from the previous slides

$$\hat{x} = \left(A^T A + h^2 I\right)^{-1} A^T y$$

Formula first derived by Tikhonov in 1963

- "Solution of incorrectly formulated problems and the regularization method," Soviet Math. Dokl. 4 (1963), pp. 1035–1038.

Here the Condition Number

It is

$$cond\left(A^TA + h^2I\right) \leq \frac{h^2 + \|A\|^2}{h^2}$$

Therefore, we have from the previous slides

$$\widehat{x} = \left(A^TA + h^2I\right)^{-1} A^Ty$$

Formula first derived by Tikhonov in 1963

- "Solution of incorrectly formulated problems and the regularization method," Soviet Math. Dokl. 4 (1963), pp. 1035–1038.

Here the Condition Number

It is

$$cond\left(A^T A + h^2 I\right) \leq \frac{h^2 + \|A\|^2}{h^2}$$

Therefore, we have from the previous slides

$$\widehat{x} = \left(A^T A + h^2 I\right)^{-1} A^T y$$

Formula first derived by Tikhonov in 1963

- "Solution of incorrectly formulated problems and the regularization method," Soviet Math. Dokl. 4 (1963), pp. 1035–1038.

Cinvestav

Finally

Corresponds to the family of approximate inverses (Tikhonov Regularization)

$$C_h = \left(A^T A + h^2 I\right)^{-1} A^T$$

Outline

A Classic Example, The Finite-Dimensional Case

Given a Matrix K of $N \times N$

- with decomposition

$$K = Q\Sigma Q^t$$

- Such that $QQ^T = I$

Where

- Σ is the matrix $diag\left(\sigma_1, \sigma_2, ..., \sigma_N\right)$ of eigenvalues with $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_N$
- $Q = \begin{bmatrix} q_1 & q_2 & \cdots & q_N \end{bmatrix}$ the corresponding eigenvectors.

Then, it is possible to write the following estimation

$$\hat{x} = K^{-1}Y = Q\Sigma^{-1}Q^T y = \sum_{i=1}^{n} \frac{1}{\sigma_i} \left\langle q_i, Y \right\rangle q_i$$

A Classic Example, The Finite-Dimensional Case

Given a Matrix K of $N \times N$

- with decomposition

$$K = Q\Sigma Q^t$$

- Such that $QQ^T = I$

Where

- Σ is the matrix $diag\left(\sigma_1, \sigma_2, ..., \sigma_N\right)$ of eigenvalues with $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_N$
- $Q = \begin{bmatrix} q_1 & q_2 & \cdots & q_N \end{bmatrix}$ the corresponding eigenvectors.

Then, it is possible to write the following estimation

$$\hat{x} = K^{-1}Y = Q\Sigma^{-1}Q^T y = \sum_{i=1}^{n} \frac{1}{\sigma_i} \langle q_i, Y \rangle q_i$$

A Classic Example, The Finite-Dimensional Case

Given a Matrix K of $N \times N$

- with decomposition

$$K = Q\Sigma Q^t$$

- Such that $QQ^T = I$

Where

- Σ is the matrix $diag\left(\sigma_1, \sigma_2, ..., \sigma_N\right)$ of eigenvalues with $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_N$
- $Q = \begin{bmatrix} q_1 & q_2 & \cdots & q_N \end{bmatrix}$ the corresponding eigenvectors.

Then, it is possible to write the following estimation

$$\widehat{\boldsymbol{x}} = K^{-1}Y = Q\Sigma^{-1}Q^T\boldsymbol{y} = \sum_{i=1}^{n} \frac{1}{\sigma_i} \left\langle q_i, Y \right\rangle q_i$$

Therefore

If we start to see really small σ_i, the solution will be unstable

- It is more, if there are zero eigenvalues, the matrix will be impossible to invert.

Clearly, the coefficients of $\hat{x}$ will go infinity

$$x_i = \frac{1}{\sigma_i}\langle q_i, Y\rangle \to \infty$$

- Or Statistical High Variance...

Therefore

If we start to see really small σ_i, the solution will be unstable

- It is more, if there are zero eigenvalues, the matrix will be impossible to invert.

Clearly, the coefficients of $\widehat{x}$ will go infinity

$$x_i = \frac{1}{\sigma_i} \langle q_i, Y \rangle \rightarrow \infty$$

- Or Statistical High Variance...

A Classic By Tikhonov

Add an extra term λ to avoid such problems

$$\widehat{\boldsymbol{x}} = (K + n\lambda I)^{-1} Y = Q\Sigma^{-1}Q^T\boldsymbol{y}$$

Again simple linear algebra

- The eigenvalues are padded by the same value, and we do not care about the effect in the eigenvectors given that we care only in the directions!!!

A Classic By Tikhonov

Add an extra term λ to avoid such problems

$$\widehat{\boldsymbol{x}} = (K + n\lambda I)^{-1} Y = Q\Sigma^{-1}Q^{T}\boldsymbol{y}$$

Again simple linear algebra

- The eigenvalues are padded by the same value, and we do not care about the effect in the eigenvectors given that we care only in the directions!!!

Thus

If we rewrite the equations

$$\widehat{\boldsymbol{x}} = Q\left(\Sigma + n\lambda I\right)^{-1} Q^T \boldsymbol{y} = \sum_{i=1}^{n} \frac{1}{\sigma_i + n\lambda} \left\langle q_i, Y \right\rangle q_i$$

Actually, regularization filters out the undesired components

- If $\sigma_i \gg \lambda n$ then $\frac{1}{\sigma_i + n\lambda} \sim \frac{1}{\sigma_i}$
- If $\sigma_i \ll \lambda n$ then $\frac{1}{\sigma_i + n\lambda} \sim \frac{1}{n\lambda}$

Thus

If we rewrite the equations

$$\widehat{\boldsymbol{x}} = Q\left(\Sigma + n\lambda I\right)^{-1} Q^T \boldsymbol{y} = \sum_{i=1}^{n} \frac{1}{\sigma_i + n\lambda} \left\langle q_i, Y \right\rangle q_i$$

Actually, regularization filters out the undesired components

- If $\sigma_i \gg \lambda n$ then $\frac{1}{\sigma_i + n\lambda} \sim \frac{1}{\sigma_i}$
- If $\sigma_i \ll \lambda n$ then $\frac{1}{\sigma_i + n\lambda} \sim \frac{1}{n\lambda}$

In a more general setup

Let be $G_\lambda(\sigma)$ a regularization function for the eigenvalues, we can then decompose K as

$$G_\lambda(K) = QG_\lambda(\sigma)Q^T$$

Therefore our estimation, finishes as

$$G_\lambda(K)y = \sum_{i=1}^{n} G_\lambda(\sigma)\langle q_i, Y\rangle q_i$$

Cinvestav

In a more general setup

Let be $G_{\lambda}(\sigma)$ a regularization function for the eigenvalues, we can then decompose K as

$$G_{\lambda}(K) = QG_{\lambda}(\sigma) Q^{T}$$

Therefore our estimation, finishes as

$$G_{\lambda}(K)\boldsymbol{y} = \sum_{i=1}^{n} G_{\lambda}(\sigma) \langle q_i, Y \rangle q_i$$

Cinvestav

Clearly

For Tikhonov

$$G_{\lambda}\left(\sigma\right) = \frac{1}{\sigma_i + n\lambda}$$

Remarks

First

- In the inverse problems literature, many algorithms are known besides Tikhonov regularization.

These algorithms are defined by a suitable G

- They are not necessarily based on **Regularized Empirical Risk Minimization** (ERM):

$$R_{emp}(f) = \frac{1}{n}\sum_{i=1}^{n} L(f(x_i), y_i)$$

- However, they perform spectral regularization (Eigenvalue Based Regularization).

Remarks

First

- In the inverse problems literature, many algorithms are known besides Tikhonov regularization.

These algorithms are defined by a suitable G

- They are not necessarily based on **Regularized Empirical Risk Minimization** (ERM):

$$R_{emp}\left(f\right) = \frac{1}{n}\sum_{i=1}^{n} L\left(f\left(x_i\right), y_i\right)$$

- However, they perform spectral regularization (Eigenvalue Based Regularization).

Spectral Filtering

Examples

1. Gradient Descent (or Landweber Iteration or L_2 Boosting)
2. ν-accelerated Landweber
3. Iterated Tikhonov Regularization
4. Truncated Singular Value Decomposition (TSVD)
5. Principle Component Regression (PCR)

Outline

Cinvestav

The Landweber Iteration

The Landweber iteration or Landweber algorithm

- It is an algorithm to solve ill-posed linear inverse problems

It is quite old...

- The method was first proposed in the 1950s by Louis Landweber,

Remarks

- When A is nonsingular, then an explicit solution is $x = A^{-1}y$

The Landweber Iteration

The Landweber iteration or Landweber algorithm

- It is an algorithm to solve ill-posed linear inverse problems

It is quite old...

- The method was first proposed in the 1950s by Louis Landweber,

Remarks

- When A is nonsingular, then an explicit solution is $x = A^{-1}y$

The Landweber Iteration

The Landweber iteration or Landweber algorithm

- It is an algorithm to solve ill-posed linear inverse problems

It is quite old...

- The method was first proposed in the 1950s by Louis Landweber,

Remarks

- When A is nonsingular, then an explicit solution is $x = A^{-1}y$

Cinvestav

Therefore

The Landweber algorithm is an attempt to regularize the problem

- The algorithm tries to solve the minimization

$$\min_{\boldsymbol{w}} \frac{\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|_2^2}{2}$$

Using the update

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k + \eta \boldsymbol{X}^T \left(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}_k\right)$$

- where $0 < \eta < 2 \left\|\boldsymbol{X}^T \boldsymbol{X}\right\|_2^{-1} = 2\sigma$

This is given by the taking in account

$$\phi\left(\boldsymbol{w}\right) = \frac{\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|_2^2}{2}$$

Therefore

The Landweber algorithm is an attempt to regularize the problem

- The algorithm tries to solve the minimization

$$\min_{\boldsymbol{w}} \frac{\left\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\right\|_2^2}{2}$$

Using the update

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k + \eta \boldsymbol{X}^T \left(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}_k\right)$$

- where $0 < \eta < 2\left\|\boldsymbol{X}^T\boldsymbol{X}\right\|_2^{-1} = 2\sigma$

This is given by the taking in account

$$\phi\left(\boldsymbol{w}\right) = \frac{\left\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\right\|_2^2}{2}$$

Therefore

The Landweber algorithm is an attempt to regularize the problem

- The algorithm tries to solve the minimization

$$\min_{\boldsymbol{w}} \frac{\|\boldsymbol{y}-\boldsymbol{X}\boldsymbol{w}\|_2^2}{2}$$

Using the update

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k + \eta \boldsymbol{X}^T \left(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}_k\right)$$

- where $0 < \eta < 2 \left\|\boldsymbol{X}^T\boldsymbol{X}\right\|_2^{-1} = 2\sigma$

This is given by the taking in account

$$\phi\left(\boldsymbol{w}\right) = \frac{\|\boldsymbol{y}-\boldsymbol{X}\boldsymbol{w}\|_2^2}{2}$$

Then

It is possible to show that the gradient of is $\phi(\boldsymbol{w})$

$$\phi(\boldsymbol{w}) = -\boldsymbol{X}^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}_k)$$

Therefore

- Each step in Landweber's method is a step in the direction of steepest descent.

Then

It is possible to show that the gradient of is $\phi(\boldsymbol{w})$

$$\phi(\boldsymbol{w}) = -\boldsymbol{X}^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}_k)$$

Therefore

- Each step in Landweber's method is a step in the direction of steepest descent.

Outline

Cinvestav

Given that the Canonical Solution has problems

We can develop a more robust algorithm

Using the Gradient Descent Idea

Basically, The Gradient Descent

It uses the change in the surface of the cost function to obtain a direction of improvement.

Cinvestav

Given that the Canonical Solution has problems

We can develop a more robust algorithm

Using the Gradient Descent Idea

Basically, The Gradient Descent

It uses the change in the surface of the cost function to obtain a direction of improvement.

Gradient Descent

The basic procedure is as follow

1. Start with a random weight vector $\boldsymbol{w}(1)$.
2. Compute the gradient vector $\nabla J(\boldsymbol{w}(1))$.
3. Obtain value $w(2)$ by moving from $w(1)$ in the direction of the steepest descent:

$$\boldsymbol{w}(k+1) = \boldsymbol{w}(k) - \eta(k) \nabla J(\boldsymbol{w}(k)) \quad (1)$$

$\eta(k)$ is a positive scale factor or learning rate!!!

Gradient Descent

The basic procedure is as follow

1. Start with a random weight vector $\boldsymbol{w}(1)$.
2. Compute the gradient vector $\nabla J(\boldsymbol{w}(1))$.
3. Obtain value $w(2)$ by moving from $w(1)$ in the direction of the steepest descent:

$$\boldsymbol{w}(k+1) = \boldsymbol{w}(k) - \eta(k) \nabla J(\boldsymbol{w}(k)) \quad (1)$$

$\eta(k)$ is a positive scale factor or learning rate!!!

Gradient Descent

The basic procedure is as follow

1. Start with a random weight vector $\boldsymbol{w}(1)$.
2. Compute the gradient vector $\nabla J(\boldsymbol{w}(1))$.
3. Obtain value $\boldsymbol{w}(2)$ by moving from $\boldsymbol{w}(1)$ in the direction of the steepest descent:

$$\boldsymbol{w}(k+1) = \boldsymbol{w}(k) - \eta(k)\nabla J(\boldsymbol{w}(k)) \quad (1)$$

$\eta(k)$ is a positive scale factor or learning rate!!!

Gradient Descent

The basic procedure is as follow

1. Start with a random weight vector $\boldsymbol{w}(1)$.
2. Compute the gradient vector $\nabla J(\boldsymbol{w}(1))$.
3. Obtain value $\boldsymbol{w}(2)$ by moving from $\boldsymbol{w}(1)$ in the direction of the steepest descent:

$$\boldsymbol{w}(k+1) = \boldsymbol{w}(k) - \eta(k) \nabla J(\boldsymbol{w}(k)) \tag{1}$$

$\eta(k)$ is a positive scale factor or learning rate!!!

Cinvestav

Gradient Descent

The basic procedure is as follow

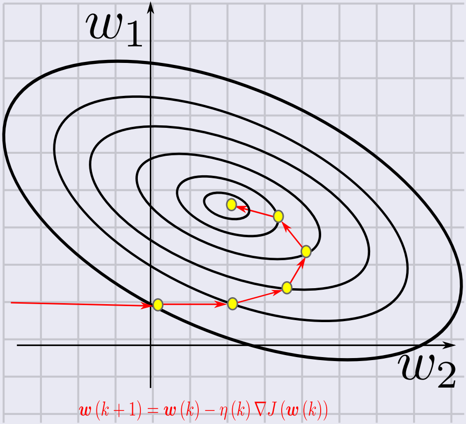
1. Start with a random weight vector $\boldsymbol{w}(1)$.
2. Compute the gradient vector $\nabla J(\boldsymbol{w}(1))$.
3. Obtain value $\boldsymbol{w}(2)$ by moving from $\boldsymbol{w}(1)$ in the direction of the steepest descent:

$$\boldsymbol{w}(k+1) = \boldsymbol{w}(k) - \eta(k) \nabla J(\boldsymbol{w}(k)) \quad (1)$$

$\eta(k)$ is a positive scale factor or learning rate!!!

Geometrically

We have the following



Outline

For our full regularized equation

We have

$$J(\boldsymbol{w}) = \frac{1}{2}\sum_{i=1}^{N}\left(y_i - \sum_{j=1}^{d+1} x_j^i w_j\right)^2 + \frac{\lambda}{2}\sum_{j=1}^{d+1} w_j^2 \tag{2}$$

Then, for each w_j

$$\frac{dJ(\boldsymbol{w})}{dw_j} = -\sum_{i=1}^{N}\left[\left(y_i - \sum_{j=1}^{d+1} x_j^i w_j\right) x_j^i\right] + \lambda w_j \tag{3}$$

Therefore

$$\nabla J(\boldsymbol{w}(k)) = \begin{pmatrix} -\sum_{i=1}^{N}\left[\left(y_i - \sum_{j=1}^{d+1} x_j^i w_j\right) x_1^i\right] + \lambda w_1 \\ \vdots \\ -\sum_{i=1}^{N}\left[\left(y_i - \sum_{j=1}^{d+1} x_j^i w_j\right) x_{d+1}^i\right] + \lambda w_{d+1} \end{pmatrix}$$

For our full regularized equation

We have

$$J(\boldsymbol{w}) = \frac{1}{2}\sum_{i=1}^{N}\left(y_i - \sum_{j=1}^{d+1} x_j^i w_j\right)^2 + \frac{\lambda}{2}\sum_{j=1}^{d+1} w_j^2 \tag{2}$$

Then, for each w_j

$$\frac{dJ(\boldsymbol{w})}{dw_j} = -\sum_{i=1}^{N}\left[\left(y_i - \sum_{j=1}^{d+1} x_j^i w_j\right) x_j^i\right] + \lambda w_j \tag{3}$$

Therefore

$$\nabla J(w(k)) = \begin{pmatrix} -\sum_{i=1}^{N}\left[\left(y_i - \sum_{j=1}^{d+1} x_j^i w_j\right) x_1^i\right] + \lambda w_1 \\ \vdots \\ -\sum_{i=1}^{N}\left[\left(y_i - \sum_{j=1}^{d+1} x_j^i w_j\right) x_{d+1}^i\right] + \lambda w_{d+1} \end{pmatrix}$$

For our full regularized equation

We have

$$J(\boldsymbol{w}) = \frac{1}{2}\sum_{i=1}^{N}\left(y_i - \sum_{j=1}^{d+1} x_j^i w_j\right)^2 + \frac{\lambda}{2}\sum_{j=1}^{d+1} w_j^2 \quad (2)$$

Then, for each w_j

$$\frac{dJ(\boldsymbol{w})}{dw_j} = -\sum_{i=1}^{N}\left[\left(y_i - \sum_{j=1}^{d+1} x_j^i w_j\right) x_j^i\right] + \lambda w_j \quad (3)$$

Therefore

$$\nabla J(\boldsymbol{w}(k)) = \begin{pmatrix} -\sum_{i=1}^{N}\left[\left(y_i - \sum_{j=1}^{d+1} x_j^i w_j\right) x_1^i\right] + \lambda w_1 \\ \vdots \\ -\sum_{i=1}^{N}\left[\left(y_i - \sum_{j=1}^{d+1} x_j^i w_j\right) x_{d+1}^i\right] + \lambda w_{d+1} \end{pmatrix}$$

Outline

Algorithm

Gradient Decent

1. Initialize $\boldsymbol{w}$, criterion θ, $\eta(\cdot)$, $k = 0$
2. do $k = k + 1$
3. $$\boldsymbol{w}(k) = \boldsymbol{w}(k-1) - \eta(k) \nabla J(\boldsymbol{w}(k-1))$$
4. until $\eta(k) \nabla J(\boldsymbol{w}(k)) < \theta$
5. return $\boldsymbol{w}$

- If $\eta(k)$ is too large, correction will overshoot and can even diverge!!!

Algorithm

Gradient Decent

1. Initialize $\boldsymbol{w}$, criterion θ, $\eta(\cdot)$, $k = 0$
2. do $k = k + 1$
3. $\qquad \boldsymbol{w}(k) = \boldsymbol{w}(k-1) - \eta(k)\nabla J(\boldsymbol{w}(k-1))$
4. until $\eta(k)\nabla J(\boldsymbol{w}(k)) < \theta$
5. return $\boldsymbol{w}$

Problem!!! How to choose the learning rate?

- If $\eta(k)$ is too small, convergence is quite slow!!!
- If $\eta(k)$ is too large, correction will overshot and can even diverge!!!

Cinvestav

Algorithm

Gradient Decent

1. Initialize $\boldsymbol{w}$, criterion θ, $\eta(\cdot)$, $k = 0$
2. do $k = k + 1$
3. $$\boldsymbol{w}(k) = \boldsymbol{w}(k-1) - \eta(k) \nabla J(\boldsymbol{w}(k-1))$$
4. until $\eta(k) \nabla J(\boldsymbol{w}(k)) < \theta$
5. return $\boldsymbol{w}$

Problem!!! How to choose the learning rate?

- If $\eta(k)$ is too small, convergence is quite slow!!!
- If $\eta(k)$ is too large, correction will overshot and can even diverge!!!

Algorithm

Gradient Decent

1. Initialize $\boldsymbol{w}$, criterion θ, $\eta(\cdot)$, $k = 0$
2. do $k = k + 1$
3. $$\boldsymbol{w}(k) = \boldsymbol{w}(k-1) - \eta(k) \nabla J(\boldsymbol{w}(k-1))$$
4. until $\eta(k) \nabla J(\boldsymbol{w}(k)) < \theta$
5. return $\boldsymbol{w}$

Problem!!! How to choose the learning rate?

- If $\eta(k)$ is too small, convergence is quite slow!!!
- If $\eta(k)$ is too large, correction will overshot and can even diverge!!!

Algorithm

Gradient Decent

1. Initialize $\boldsymbol{w}$, criterion θ, $\eta(\cdot)$, $k = 0$
2. do $k = k + 1$
3. $\qquad \boldsymbol{w}(k) = \boldsymbol{w}(k-1) - \eta(k) \nabla J(\boldsymbol{w}(k-1))$
4. until $\eta(k) \nabla J(\boldsymbol{w}(k)) < \theta$
5. return $\boldsymbol{w}$

Problem!!! How to choose the learning rate?

- If $\eta(k)$ is too small, convergence is quite slow!!!
- If $\eta(k)$ is too large, correction will overshot and can even diverge!!!

Algorithm

Gradient Decent

1. Initialize $\boldsymbol{w}$, criterion θ, $\eta(\cdot)$, $k = 0$
2. do $k = k + 1$
3. $\quad\quad \boldsymbol{w}(k) = \boldsymbol{w}(k-1) - \eta(k)\nabla J(\boldsymbol{w}(k-1))$
4. until $\eta(k)\nabla J(\boldsymbol{w}(k)) < \theta$
5. return $\boldsymbol{w}$

Problem!!! How to choose the learning rate?

- If $\eta(k)$ is too small, convergence is quite slow!!!
- If $\eta(k)$ is too large, correction will overshot and can even diverge!!!

Algorithm

Gradient Decent

1. Initialize $\boldsymbol{w}$, criterion θ, $\eta(\cdot)$, $k = 0$
2. do $k = k + 1$
3. $\qquad \boldsymbol{w}(k) = \boldsymbol{w}(k-1) - \eta(k) \nabla J(\boldsymbol{w}(k-1))$
4. until $\eta(k) \nabla J(\boldsymbol{w}(k)) < \theta$
5. return $\boldsymbol{w}$

Problem!!! How to choose the learning rate?

- If $\eta(k)$ is too small, convergence is quite slow!!!
- If $\eta(k)$ is too large, correction will overshot and can even diverge!!!

Cinvestav

Outline

Cinvestav

Using the Taylor's second-order expansion around value $\boldsymbol{w}(\boldsymbol{k})$

We do the following

$$J(\boldsymbol{w}) = J(\boldsymbol{w}(k)) + \nabla J^{T}(\boldsymbol{w} - \boldsymbol{w}(k)) + \frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}(k))^{T} \boldsymbol{H} (\boldsymbol{w} - \boldsymbol{w}(k)) \tag{4}$$

Remark: This is know as Taylor's Second Order expansion!!!

Using the Taylor's second-order expansion around value $\boldsymbol{w}(\boldsymbol{k})$

We do the following

$$J(\boldsymbol{w}) = J(\boldsymbol{w}(k)) + \nabla J^T (\boldsymbol{w} - \boldsymbol{w}(k)) + \frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}(k))^T \boldsymbol{H} (\boldsymbol{w} - \boldsymbol{w}(k)) \tag{4}$$

Remark: This is know as Taylor's Second Order expansion!!!

Here, we have

- ∇J is the vector of partial derivatives $\frac{\partial J}{\partial w_i}$ evaluated at $\boldsymbol{w}(k)$.
- $\boldsymbol{H}$ is the Hessian matrix of second partial derivatives $\frac{\partial^2 J}{\partial w_i \partial w_j}$ evaluated at $\boldsymbol{w}(k)$.

Using the Taylor's second-order expansion around value $\boldsymbol{w}(\boldsymbol{k})$

We do the following

$$J(\boldsymbol{w}) = J(\boldsymbol{w}(k)) + \nabla J^T (\boldsymbol{w} - \boldsymbol{w}(k)) + \frac{1}{2} (\boldsymbol{w} - \boldsymbol{w}(k))^T \boldsymbol{H} (\boldsymbol{w} - \boldsymbol{w}(k)) \quad (4)$$

Remark: This is know as Taylor's Second Order expansion!!!

Here, we have

- ∇J is the vector of partial derivatives $\frac{\partial J}{\partial w_i}$ evaluated at $\boldsymbol{w}(k)$.
- $\boldsymbol{H}$ is the Hessian matrix of second partial derivatives $\frac{\partial^2 J}{\partial w_i \partial w_j}$ evaluated at $\boldsymbol{w}(k)$.

Using the Taylor's second-order expansion around value $\boldsymbol{w}(\boldsymbol{k})$

We do the following

$$J(\boldsymbol{w}) = J(\boldsymbol{w}(k)) + \nabla J^T(\boldsymbol{w} - \boldsymbol{w}(k)) + \frac{1}{2}(\boldsymbol{w} - \boldsymbol{w}(k))^T \boldsymbol{H}(\boldsymbol{w} - \boldsymbol{w}(k)) \tag{4}$$

Remark: This is know as Taylor's Second Order expansion!!!

Here, we have

- ∇J is the vector of partial derivatives $\frac{\partial J}{\partial w_i}$ evaluated at $\boldsymbol{w}(k)$.
- $\boldsymbol{H}$ is the Hessian matrix of second partial derivatives $\frac{\partial^2 J}{\partial w_i \partial w_j}$ evaluated at $\boldsymbol{w}(k)$.

Cinvestav

Then

We substitute (Eq. 1) into (Eq. 4)

$$\boldsymbol{w}(k+1) - \boldsymbol{w}(k) = \eta(k)\,\nabla J(\boldsymbol{w}(k)) \tag{5}$$

We have then

$$\begin{aligned} J(\boldsymbol{w}(k+1)) \cong & J(\boldsymbol{w}(k)) + \nabla J^T(-\eta(k)\nabla J(\boldsymbol{w}(k))) + \dots \\ & \frac{1}{2}(-\eta(k)\nabla J(\boldsymbol{w}(k)))^T \boldsymbol{H}(-\eta(k)\nabla J(\boldsymbol{w}(k))) \end{aligned}$$

Finally, we have

$$J(\boldsymbol{w}(k+1)) \cong J(\boldsymbol{w}(k)) - \eta(k)\|\nabla J\|^2 + \frac{1}{2}\eta^2(k)\nabla J^T \boldsymbol{H}\nabla J \tag{6}$$

Cinvestav

Then

We substitute (Eq. 1) into (Eq. 4)

$$\boldsymbol{w}(k+1) - \boldsymbol{w}(k) = \eta(k) \nabla J(\boldsymbol{w}(k)) \tag{5}$$

We have then

$$\begin{aligned} J(\boldsymbol{w}(k+1)) \cong & J(\boldsymbol{w}(k)) + \nabla J^T (-\eta(k) \nabla J(\boldsymbol{w}(k))) + \ldots \\ & \frac{1}{2} (-\eta(k) \nabla J(\boldsymbol{w}(k)))^T \boldsymbol{H} (-\eta(k) \nabla J(\boldsymbol{w}(k))) \end{aligned}$$

Finally, we have

$$J(\boldsymbol{w}(k+1)) \cong J(\boldsymbol{w}(k)) - \eta(k) \|\nabla J\|^2 + \frac{1}{2}\eta^2(k) \nabla J^T \boldsymbol{H} \nabla J \tag{6}$$

Cinvestav

Then

We substitute (Eq. 1) into (Eq. 4)

$$\boldsymbol{w}(k+1) - \boldsymbol{w}(k) = \eta(k) \nabla J(\boldsymbol{w}(k)) \tag{5}$$

We have then

$$\begin{aligned} J(\boldsymbol{w}(k+1)) \cong & J(\boldsymbol{w}(k)) + \nabla J^T(-\eta(k) \nabla J(\boldsymbol{w}(k))) + \ldots \\ & \frac{1}{2}(-\eta(k) \nabla J(\boldsymbol{w}(k)))^T \boldsymbol{H}(-\eta(k) \nabla J(\boldsymbol{w}(k))) \end{aligned}$$

Finally, we have

$$J(\boldsymbol{w}(k+1)) \cong J(\boldsymbol{w}(k)) - \eta(k) \|\nabla J\|^2 + \frac{1}{2}\eta^2(k) \nabla J^T \boldsymbol{H} \nabla J \tag{6}$$

Derive with respect to $\eta(k)$ and make the result equal to zero

We have then

$$-\|\nabla J\|^2 + \eta(k)\nabla J^T \boldsymbol{H} \nabla J = 0 \tag{7}$$

Finally

$$\eta(k) = \frac{\|\nabla J\|^2}{\nabla J^T \boldsymbol{H} \nabla J} \tag{8}$$

Remark: This is the optimal step size!!!

Problem!!!

Calculating $\boldsymbol{H}$ can be quite expansive!!!

Derive with respect to $\eta(k)$ and make the result equal to zero

We have then

$$-\|\nabla J\|^2 + \eta(k)\,\nabla J^T \boldsymbol{H} \nabla J = 0 \tag{7}$$

Finally

$$\eta(k) = \frac{\|\nabla J\|^2}{\nabla J^T \boldsymbol{H} \nabla J} \tag{8}$$

Remark This is the optimal step size!!!

Problem!!!

Calculating $\boldsymbol{H}$ can be quite expansive!!!

Derive with respect to $\eta(k)$ and make the result equal to zero

We have then

$$-\|\nabla J\|^2 + \eta(k) \nabla J^T \boldsymbol{H} \nabla J = 0 \quad (7)$$

Finally

$$\eta(k) = \frac{\|\nabla J\|^2}{\nabla J^T \boldsymbol{H} \nabla J} \quad (8)$$

Remark This is the optimal step size!!!

Problem!!!

Calculating $\boldsymbol{H}$ can be quite expansive!!!

We can have an adaptive linear search!!!

We can use the idea of having everything fixed, but $\eta(k)$

Then, we can have the following function

$$f(\eta(k)) = J(\boldsymbol{w}(k) - \eta(k)\nabla J(\boldsymbol{w}(k)))$$

- We can optimized using linear search methods

We can have an adaptive linear search!!!

We can use the idea of having everything fixed, but $\eta(k)$

Then, we can have the following function

$$f(\eta(k)) = J(\boldsymbol{w}(k) - \eta(k)\nabla J(\boldsymbol{w}(k)))$$

- We can optimized using linear search methods

Linear Search Methods

- Backtracking linear search
- Bisection method
- Golden ratio
- Etc.

We can have an adaptive linear search!!!

We can use the idea of having everything fixed, but $\eta(k)$

Then, we can have the following function

$$f(\eta(k)) = J(\boldsymbol{w}(k) - \eta(k)\nabla J(\boldsymbol{w}(k)))$$

- We can optimized using linear search methods

Linear Search Methods

- Backtracking linear search
- Bisection method
- Golden ratio
- Etc.

We can have an adaptive linear search!!!

We can use the idea of having everything fixed, but $\eta(k)$

Then, we can have the following function

$$f(\eta(k)) = J(\boldsymbol{w}(k) - \eta(k)\nabla J(\boldsymbol{w}(k)))$$

- We can optimized using linear search methods

Linear Search Methods

- Backtracking linear search
- Bisection method
- Golden ratio
- Etc.

We can have an adaptive linear search!!!

We can use the idea of having everything fixed, but $\eta(k)$

Then, we can have the following function

$f(\eta(k)) = J(\boldsymbol{w}(k) - \eta(k)\nabla J(\boldsymbol{w}(k)))$

- We can optimized using linear search methods

Linear Search Methods

- Backtracking linear search
- Bisection method
- Golden ratio
- Etc.

Cinvestav

We can have an adaptive linear search!!!

We can use the idea of having everything fixed, but $\eta(k)$

Then, we can have the following function

$$f(\eta(k)) = J(\boldsymbol{w}(k) - \eta(k)\nabla J(\boldsymbol{w}(k)))$$

- We can optimized using linear search methods

Linear Search Methods

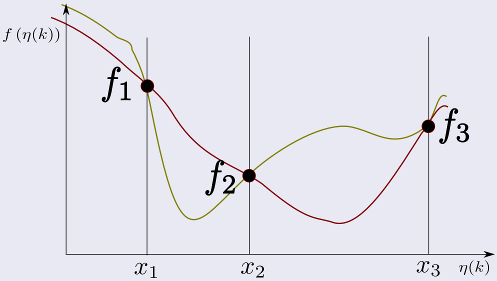
- Backtracking linear search
- Bisection method
- Golden ratio
- Etc.

Outline

Cinvestav

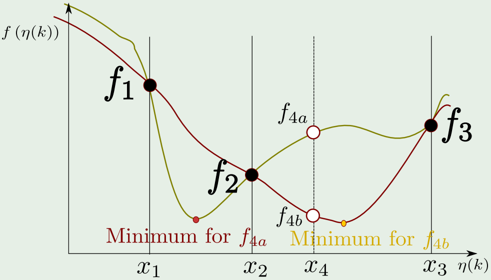
Gold Section

We have $f(\eta(k)) = J(\boldsymbol{w}(k) - \eta(k)\nabla J(\boldsymbol{w}(k)))$



Golden Section

Thus the idea is to use an evaluation f_4 to decide which subsection to drop



What is the Golden Ratio Idea?

Basically, given an interval $[x_1, x_3]$

Then, we select a point x_2 and x_3 such that we have a two possible intervals of search for the minimum

1. $[x_1, x_4]$
2. $[x_2, x_3]$

The Golden Linear Search requires these intervals be equal!!!

If they are not,

- You could run to a series of search wider intervals slowing down the rate of convergence.

What is the Golden Ratio Idea?

Basically, given an interval $[x_1, x_3]$

Then, we select a point x_2 and x_3 such that we have a two possible intervals of search for the minimum

1. $[x_1, x_4]$
2. $[x_2, x_3]$

The Golden Linear Search requires these intervals be equal!!!

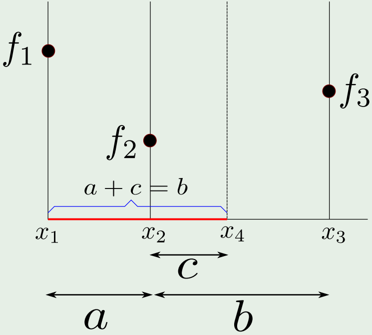
If they are not,

- You could run to a series of search wider intervals slowing down the rate of convergence.

Cinvestav

How?

By the equality $b = a + c$



Therefore

We have the following question?

Where do you place x_2? Thus you can generate x_4

You want to avoid

- x_2 to close to x_1 or x_3

Therefore

We have the following question?

Where do you place x_2? Thus you can generate x_4

You want to avoid

- x_2 to close to x_1 or x_3

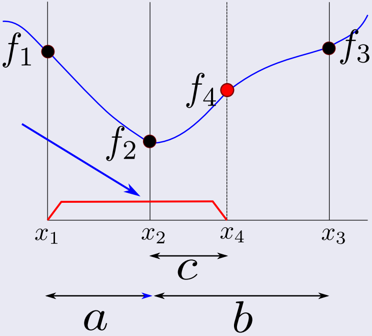
The process is as follow

We define

- $f_1 = f(x_1)$
- $f_2 = f(x_2)$
- $f_3 = f(x_3)$
- $f_4 = f(x_4)$

Two Cases

If $f_2 < f_4$ then the minimum lies between x_1 and x_4 and the new triplet is x_1, x_2 and x_4.



Here,we have the realization that

We have interval size reduction

$$x_4 - x_1 = \varphi\left(x_3 - x_1\right) \longmapsto x_4 = x_1 + \varphi x_3 - \varphi x_1$$

Then

$$x_4 = \left(1 - \varphi\right) x_1 + \varphi x_3$$

Cinvestav

Here,we have the realization that

We have interval size reduction

$$x_4 - x_1 = \varphi\left(x_3 - x_1\right) \longmapsto x_4 = x_1 + \varphi x_3 - \varphi x_1$$

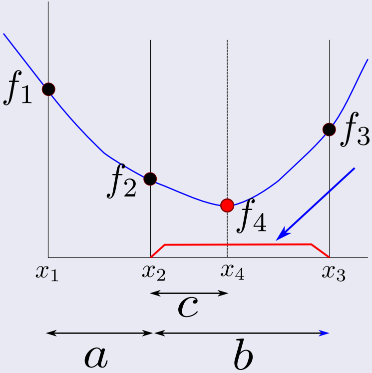
Then

$$x_4 = \left(1 - \varphi\right) x_1 + \varphi x_3$$

Cinvestav

Two Cases

If $f_4 < f_2$ then the minimum lies between x_2 and x_3 and the new triplet is x_2, x_4 and x_3.



Then

We want

$$x_3 - x_2 = \varphi\left(x_3 - x_1\right) \longmapsto -x_2 = \varphi x_3 - \varphi x_1 - x_3$$

Therefore

$$x_2 = \varphi x_1 + (1-\varphi) x_3$$

Thus, once we obtain φ, we get x_2 and x_4

- For this, we make the following assumption $[x_1, x_3] = [0, 1]$

Then

We want

$$x_3 - x_2 = \varphi\left(x_3 - x_1\right) \longmapsto -x_2 = \varphi x_3 - \varphi x_1 - x_3$$

Therefore

$$x_2 = \varphi x_1 + (1 - \varphi)x_3$$

Thus, once we obtain φ, we get x_2 and x_4

- For this, we make the following assumption $[x_1, x_3] = [0, 1]$

Then

We want

$$x_3 - x_2 = \varphi\left(x_3 - x_1\right) \longmapsto -x_2 = \varphi x_3 - \varphi x_1 - x_3$$

Therefore

$$x_2 = \varphi x_1 + (1-\varphi)x_3$$

Thus, once we obtain φ, we get x_2 and x_4

- For this, we make the following assumption $[x_1, x_3] = [0, 1]$

Therefore

If we have $f_2 < f_4$

$$x_2 = 1 - \varphi$$

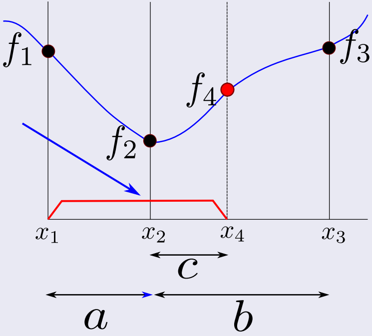
Then, if we have the new function evaluation at the left of x_2

Therefore

If we have $f_2 < f_4$

$$x_2 = 1 - \varphi$$

Then, if we have the new function evaluation at the left of x_2



With a Little Algebra

Then, x_2 is between the the interval $[0, \varphi]$ and assume is a convex combination of such values

$$1 - \varphi = (1 - \varphi)\,0 + \varphi\varphi \longmapsto \boldsymbol{\varphi^2 + \varphi - 1 = 0}$$

With Solution

$$\varphi = \frac{-1 + \sqrt{5}}{2} = 0.6180$$

With a Little Algebra

Then, x_2 is between the the interval $[0, \varphi]$ and assume is a convex combination of such values

$$1 - \varphi = (1 - \varphi)\, 0 + \varphi\varphi \longmapsto \boldsymbol{\varphi^2 + \varphi - 1 = 0}$$

With Solution

$$\varphi = \frac{-1 + \sqrt{5}}{2} = 0.6180$$

Finally, we have the algorithm

Golden Ratio

INPUT: $x_1, x_3, \tau, \varphi, f$

OUTPUT: $\frac{x_3 - x_1}{2}$

1. $x_2 = \varphi x_1 + (1 - \varphi) x_3$
2. $x_4 = (1 - \varphi) x_1 + \varphi x_3$
3. while $|x_3 - x_1| > \tau (|x_2| + |x_4|)$
4. if $f(x_2) < f(x_4)$:
5. $x_3 = x_4$
6. $x_4 = x_2$
7. $x_2 = \varphi x_1 + (1 - \varphi) x_3$
8. else
9. $x_1 = x_2$
10. $x_2 = x_4$
11. $x_4 = (1 - \varphi) x_1 + \varphi x_3$
12. return $\frac{x_3 - x_1}{2}$

Finally, we have the algorithm

Golden Ratio

INPUT: $x_1, x_3, \tau, \varphi, f$

OUTPUT: $\frac{x_3 - x_1}{2}$

1. $x_2 = \varphi x_1 + (1 - \varphi) x_3$
2. $x_4 = (1 - \varphi) x_1 + \varphi x_3$
3. while $|x_3 - x_1| > \tau \left(|x_2| + |x_4|\right)$
4. if $f(x_2) < f(x_4)$:
5. $x_3 = x_4$
6. $x_4 = x_2$
7. $x_2 = \varphi x_1 + (1 - \varphi) x_3$
8. else
9. $x_1 = x_2$
10. $x_2 = x_4$
11. $x_4 = (1 - \varphi) x_1 + \varphi x_3$
12. return $\frac{x_3 - x_1}{2}$

Finally, we have the algorithm

Golden Ratio

INPUT: $x_1, x_3, \tau, \varphi, f$

OUTPUT: $\frac{x_3 - x_1}{2}$

1. $x_2 = \varphi x_1 + (1 - \varphi) x_3$
2. $x_4 = (1 - \varphi) x_1 + \varphi x_3$
3. while $|x_3 - x_1| > \tau \left(|x_2| + |x_4|\right)$
4. if $f(x_2) < f(x_4)$:
5. $x_3 = x_4$
6. $x_4 = x_2$
7. $x_2 = \varphi x_1 + (1 - \varphi) x_3$
8. else
9. $x_1 = x_2$
10. $x_2 = x_4$
11. $x_4 = (1 - \varphi) x_1 + \varphi x_3$
12. return $\frac{x_3 - x_1}{2}$

Finally, we have the algorithm

Golden Ratio

INPUT: $x_1, x_3, \tau, \varphi, f$

OUTPUT: $\frac{x_3 - x_1}{2}$

1. $x_2 = \varphi x_1 + (1-\varphi)x_3$
2. $x_4 = (1-\varphi)x_1 + \varphi x_3$
3. while $|x_3 - x_1| > \tau\left(|x_2| + |x_4|\right)$
4. if $f(x_2) < f(x_4)$:
5. $x_3 = x_4$
6. $x_4 = x_2$
7. $x_2 = \varphi x_1 + (1-\varphi)x_3$
8. else
9. $x_1 = x_2$
10. $x_2 = x_4$
11. $x_4 = (1-\varphi)x_1 + \varphi x_3$
12. return $\frac{x_3 - x_1}{2}$

Finally, we have the algorithm

Golden Ratio

INPUT: $x_1, x_3, \tau, \varphi, f$

OUTPUT: $\frac{x_3 - x_1}{2}$

$x_2 = \varphi x_1 + (1-\varphi) x_3$

$x_4 = (1-\varphi) x_1 + \varphi x_3$

while $|x_3 - x_1| > \tau \left(|x_2| + |x_4|\right)$

    if $f(x_2) < f(x_4)$:

        $x_3 = x_4$

        $x_4 = x_2$

        $x_2 = \varphi x_1 + (1-\varphi) x_3$

    else

        $x_1 = x_2$

        $x_2 = x_4$

        $x_4 = (1-\varphi) x_1 + \varphi x_3$

return $\frac{x_3 - x_1}{2}$

Finally, we have the algorithm

Golden Ratio

INPUT: $x_1, x_3, \tau, \varphi, f$

OUTPUT: $\frac{x_3 - x_1}{2}$

$x_2 = \varphi x_1 + (1-\varphi) x_3$

$x_4 = (1-\varphi) x_1 + \varphi x_3$

while $|x_3 - x_1| > \tau \left(|x_2| + |x_4|\right)$

  if $f(x_2) < f(x_4)$:

    $x_3 = x_4$

    $x_4 = x_2$

    $x_2 = \varphi x_1 + (1-\varphi) x_3$

  else

    $x_1 = x_2$

    $x_2 = x_4$

    $x_4 = (1-\varphi) x_1 + \varphi x_3$

return $\frac{x_3 - x_1}{2}$

Iteratively

Repeat the procedure!!!

Until a error threshold is reached.

Iteratively

Repeat the procedure!!!

Until a error threshold is reached.

For more, please read the paper

"SEQUENTIAL MINIMAX SEARCH FOR A MAXIMUM" by J. Kiefer

Iteratively

Repeat the procedure!!!

Until a error threshold is reached.

For more, please read the paper

"SEQUENTIAL MINIMAX SEARCH FOR A MAXIMUM" by J. Kiefer

There are better versions

Take a look

The papers at the repository.

Outline

Cinvestav

The Gauss-Markov Theorem

Given the Linear Estimation Model

$$\boldsymbol{y} = X\boldsymbol{w} + \boldsymbol{\epsilon}$$

Under the following assumptions

1. $E[\epsilon|\boldsymbol{x}] = 0$ for all $\boldsymbol{x}$ (Mean Independence).
2. $Var[\epsilon] = E\left[\epsilon\epsilon^T|\boldsymbol{x}\right] = \sigma_\epsilon^2 I_N$ (Homoskedasticity).

The Gauss-Markov Theorem states

$$\widehat{w} = \left(X^T X\right)^{-1} X^T y$$

is the Best Linear Unbiased Estimator (BLUE), if ϵ satisfies 1. and 2.!!!

Cinvestav

The Gauss-Markov Theorem

Given the Linear Estimation Model

$$\boldsymbol{y} = X\boldsymbol{w} + \boldsymbol{\epsilon}$$

Under the following assumptions

1. $E\left[\boldsymbol{\epsilon}|\boldsymbol{x}\right] = \boldsymbol{0}$ for all $\boldsymbol{x}$ (Mean Independence).
2. $Var\left[\boldsymbol{\epsilon}\right] = E\left[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T|\boldsymbol{x}\right] = \sigma_\epsilon^2 I_N$ (Homoskedasticity).

The Gauss-Markov Theorem states

$$\widehat{w} = \left(X^T X\right)^{-1} X^T y$$

is the Best Linear Unbiased Estimator (BLUE), if ϵ satisfies 1. and 2.!!!

The Gauss-Markov Theorem

Given the Linear Estimation Model

$$\boldsymbol{y} = X\boldsymbol{w} + \boldsymbol{\epsilon}$$

Under the following assumptions

1. $E\left[\boldsymbol{\epsilon}|\boldsymbol{x}\right] = \mathbf{0}$ for all $\boldsymbol{x}$ (Mean Independence).
2. $Var\left[\boldsymbol{\epsilon}\right] = E\left[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T|\boldsymbol{x}\right] = \sigma_\epsilon^2 I_N$ (Homoskedasticity).

The Gauss-Markov Theorem states

$$\widehat{\boldsymbol{w}} = \left(X^T X\right)^{-1} X^T \boldsymbol{y}$$

is the Best Linear Unbiased Estimator (BLUE), if $\boldsymbol{\epsilon}$ satisfies 1. and 2.!!!

Outline

Cinvestav

Proof

First and Fore most

- "An estimator is "best" in a class if it has smaller variance than others estimators in the same class."

Also

- We are restricting our search for estimators to the class of linear, unbiased ones

Unbiased Estimator

Given a sequence of observations $x_1, x_2, ..., x_N \sim P(X|\theta)$ then bias is the mean of the difference

$$b_d(\theta) = E[d(X) - h(\theta)]$$

with $d(X)$ is an estimator of the statistic $h(\theta)$.

Proof

First and Fore most

- "An estimator is "best" in a class if it has smaller variance than others estimators in the same class."

Also

- We are restricting our search for estimators to the class of linear, unbiased ones

Unbiased Estimator

Given a sequence of observations $x_1, x_2, ..., x_N \sim P(X|\theta)$ then bias is the mean of the difference

$$b_d(\theta) = E[d(X) - h(\theta)]$$

with $d(X)$ is an estimator of the statistic $h(\theta)$.

Proof

First and Fore most

- "An estimator is "best" in a class if it has smaller variance than others estimators in the same class."

Also

- We are restricting our search for estimators to the class of linear, unbiased ones

Unbiased Estimator

Given a sequence of observations $x_1, x_2, ..., x_N \sim P(X|\theta)$ then bias is the mean of the difference

$$b_d(\theta) = E[d(X) - h(\theta)]$$

with $d(X)$ is an estimator of the statistic $h(\theta)$.

Remark

We need to calculate estimators which have covariances

- The best estimator in a class of estimators is the one with **the "smallest" covariance matrix**

Thus

- We will look at such covariance matrix for the BLUE estimator.

Remark

We need to calculate estimators which have covariances

- The best estimator in a class of estimators is the one with **the “smallest” covariance matrix**

Thus

- We will look at such covariance matrix for the BLUE estimator.

Therefore, going back to our unbiased estimators

If $b_d(\theta) = 0$ for all values of the parameter

- Then, $d(X)$ is called an unbiased estimator.

Now, the data are the y, we are looking at estimators that are linear functions of y

$$\widetilde{\boldsymbol{w}} = \boldsymbol{m} + M\boldsymbol{y}$$

Here

- $\widetilde{\boldsymbol{w}}$ is a $k \times 1$ parameter vector
- $\boldsymbol{m}$ is a $k \times 1$ vector of constants,
- M is a $k \times N$ matrix of constants,
- The data vector $\boldsymbol{y}$ is $N \times 1$.

Therefore, going back to our unbiased estimators

If $b_d(\theta) = 0$ for all values of the parameter

- Then, $d(X)$ is called an unbiased estimator.

Now, the data are the $\boldsymbol{y}$, we are looking at estimators that are linear functions of $\boldsymbol{y}$

$$\widetilde{\boldsymbol{w}} = \boldsymbol{m} + M\boldsymbol{y}$$

Here

- $\widetilde{w}$ is a $k \times 1$ parameter vector
- m is a $k \times 1$ vector of constants,
- M is a $k \times N$ matrix of constants,
- The data vector y is $N \times 1$.

Therefore, going back to our unbiased estimators

If $b_d(\theta) = 0$ for all values of the parameter

- Then, $d(X)$ is called an unbiased estimator.

Now, the data are the $\boldsymbol{y}$, we are looking at estimators that are linear functions of $\boldsymbol{y}$

$$\widetilde{\boldsymbol{w}} = \boldsymbol{m} + M\boldsymbol{y}$$

Here

- $\widetilde{\boldsymbol{w}}$ is a $k \times 1$ parameter vector
- $\boldsymbol{m}$ is a $k \times 1$ vector of constants,
- $M$is a $k \times N$ matrix of constants,
- The data vector $\boldsymbol{y}$ is $N \times 1$.

Now

We are looking at unbiased estimators

$$E\left[\widetilde{\boldsymbol{w}}\right]=\boldsymbol{w}$$

Now

We are looking at unbiased estimators

$$E\left[\widetilde{\boldsymbol{w}}\right] = \boldsymbol{w}$$

if $\widetilde{\boldsymbol{w}}$ is to be unbiased

$$\begin{aligned} E\left[\widetilde{\boldsymbol{w}}|X\right] &= \boldsymbol{m} + ME\left[\boldsymbol{y}|X\right] \\ &= \boldsymbol{m} + ME\left[X\boldsymbol{w} + \epsilon|X\right] \\ &= \boldsymbol{m} + MX\boldsymbol{w} \end{aligned}$$

Now

We are looking at unbiased estimators

$$E\left[\widetilde{\boldsymbol{w}}\right] = \boldsymbol{w}$$

if $\widetilde{\boldsymbol{w}}$ is to be unbiased

$$\begin{aligned} E\left[\widetilde{\boldsymbol{w}}|X\right] &= \boldsymbol{m} + ME\left[\boldsymbol{y}|X\right] \\ &= \boldsymbol{m} + ME\left[X\boldsymbol{w} + \boldsymbol{\epsilon}|X\right] \\ &= \boldsymbol{m} + MX\boldsymbol{w} \end{aligned}$$

Now

We are looking at unbiased estimators

$$E\left[\widetilde{\boldsymbol{w}}\right] = \boldsymbol{w}$$

if $\widetilde{\boldsymbol{w}}$ is to be unbiased

$$\begin{aligned} E\left[\widetilde{\boldsymbol{w}}|X\right] &= \boldsymbol{m} + ME\left[\boldsymbol{y}|X\right] \\ &= \boldsymbol{m} + ME\left[X\boldsymbol{w} + \boldsymbol{\epsilon}|X\right] \\ &= \boldsymbol{m} + MX\boldsymbol{w} \end{aligned}$$

Now, we are forced

Given that we are looking for an unbiased estimator

$$\boldsymbol{m} = 0 \text{ with } MX = I_k$$

For the least squared error

$$M = \left(X^T X\right)^{-1} X^T \Longrightarrow MX = \left(X^T X\right)^{-1} X^T X = I_k$$

Looking for linear unbiased estimators requires to look for estimators as

$$\widetilde{w} = My \text{ with } MX = I_k$$

Now, we are forced

Given that we are looking for an unbiased estimator

$$\boldsymbol{m} = 0 \text{ with } MX = I_k$$

For the least squared error

$$M = \left(X^T X\right)^{-1} X^T \Longrightarrow MX = \left(X^T X\right)^{-1} X^T X = I_k$$

Looking for linear unbiased estimators requires to look for estimators as

$$\tilde{w} = My \text{ with } MX = I_k$$

Now, we are forced

Given that we are looking for an unbiased estimator

$$\boldsymbol{m} = 0 \text{ with } MX = I_k$$

For the least squared error

$$M = \left(X^T X\right)^{-1} X^T \Longmapsto MX = \left(X^T X\right)^{-1} X^T X = I_k$$

Looking for linear unbiased estimators requires to look for estimators as

$$\widetilde{\boldsymbol{w}} = M\boldsymbol{y} \text{ with } MX = I_k$$

Cinvestav

Therefore

We are looking at matrices as

$$M = \left(X^T X\right)^{-1} X^T + C$$

where C is some $k \times n$ matrix.

Therefore

We are looking at matrices as

$$M = \left(X^T X\right)^{-1} X^T + C$$

where C is some $k \times n$ matrix.

Now

$$\begin{aligned} MX &= \left[\left(X^T X\right)^{-1} X^T + C\right] X \\ &= I_k + CX = I_k \\ &\Longrightarrow CX = 0 \end{aligned}$$

Therefore

We are looking at matrices as

$$M = \left(X^T X\right)^{-1} X^T + C$$

where C is some $k \times n$ matrix.

Now

$$\begin{aligned} MX &= \left[\left(X^T X\right)^{-1} X^T + C\right] X \\ &= I_k + CX = I_k \\ &\implies CX = 0 \end{aligned}$$

Therefore

We are looking at matrices as

$$M = \left(X^T X\right)^{-1} X^T + C$$

where C is some $k \times n$ matrix.

Now

$$\begin{aligned} MX &= \left[\left(X^T X\right)^{-1} X^T + C\right] X \\ &= I_k + CX = I_k \\ &\implies CX = 0 \end{aligned}$$

Therefore

We are looking at matrices as

$$M = \left(X^T X\right)^{-1} X^T + C$$

where C is some $k \times n$ matrix.

Now

$$\begin{aligned} MX &= \left[\left(X^T X\right)^{-1} X^T + C\right] X \\ &= I_k + CX = I_k \\ &\Longrightarrow CX = 0 \end{aligned}$$

Therefore, we can compute the covariance matrix

For all alternative estimators $\widetilde{\boldsymbol{w}}$

$$\begin{aligned}\widetilde{\boldsymbol{w}} &= M\boldsymbol{y}\\ &= M\left[X\boldsymbol{w}+\epsilon\right]\\ &= \boldsymbol{w}+M\epsilon\end{aligned}$$

Therefore, we can compute the covariance matrix

For all alternative estimators $\widetilde{\boldsymbol{w}}$

$$
\begin{aligned}
\widetilde{\boldsymbol{w}} &= M\boldsymbol{y} \\
&= M\left[X\boldsymbol{w} + \boldsymbol{\epsilon}\right] \\
&= \boldsymbol{w} + M\boldsymbol{\epsilon}
\end{aligned}
$$

Therefore, the difference is $\widetilde{\boldsymbol{w}} - \boldsymbol{w} = M\boldsymbol{\epsilon}$

- And since the $\widetilde{\boldsymbol{w}}$ is unbiased, $E\left[\widetilde{\boldsymbol{w}} - \boldsymbol{w} | X\right] = 0$

Therefore, we can compute the covariance matrix

For all alternative estimators $\widetilde{\boldsymbol{w}}$

$$
\begin{aligned}
\widetilde{\boldsymbol{w}} &= M\boldsymbol{y} \\
&= M\left[X\boldsymbol{w} + \boldsymbol{\epsilon}\right] \\
&= \boldsymbol{w} + M\boldsymbol{\epsilon}
\end{aligned}
$$

Therefore, the difference is $\widetilde{\boldsymbol{w}} - \boldsymbol{w} = M\boldsymbol{\epsilon}$

- And since the $\widetilde{\boldsymbol{w}}$ is unbiased, $E\left[\widetilde{\boldsymbol{w}} - \boldsymbol{w}|X\right] = 0$

Cinvestav

Therefore, we can compute the covariance matrix

For all alternative estimators $\widetilde{\boldsymbol{w}}$

$$\begin{aligned}\widetilde{\boldsymbol{w}} &= M\boldsymbol{y}\\ &= M\left[X\boldsymbol{w}+\boldsymbol{\epsilon}\right]\\ &= \boldsymbol{w}+M\boldsymbol{\epsilon}\end{aligned}$$

Therefore, the difference is $\widetilde{\boldsymbol{w}}-\boldsymbol{w}=M\boldsymbol{\epsilon}$

- And since the $\widetilde{\boldsymbol{w}}$ is unbiased, $E\left[\widetilde{\boldsymbol{w}}-\boldsymbol{w}|X\right]=0$

We have

The Covariance Matrix

$$\begin{aligned}
E\left[\left(\widetilde{\boldsymbol{w}}-\boldsymbol{w}\right)\left(\widetilde{\boldsymbol{w}}-\boldsymbol{w}\right)^{T}|X\right] &= E\left[M\boldsymbol{\epsilon}\left(M\boldsymbol{\epsilon}\right)^{T}|X\right]\\
&= E\left[M\boldsymbol{\epsilon}\boldsymbol{\epsilon}^{T}M^{T}|X\right]\\
&= ME\left[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^{T}|X\right]M^{T}\\
&= M\sigma_{\epsilon}^{2}I_{N}M^{T}\\
&= \sigma_{\epsilon}^{2}MM^{T}
\end{aligned}$$

Cinvestav

We have

The Covariance Matrix

$$
\begin{aligned}
E\left[\left(\widetilde{\boldsymbol{w}}-\boldsymbol{w}\right)\left(\widetilde{\boldsymbol{w}}-\boldsymbol{w}\right)^{T}|X\right] &= E\left[M\boldsymbol{\epsilon}\left(M\boldsymbol{\epsilon}\right)^{T}|X\right]\\
&= E\left[M\boldsymbol{\epsilon}\boldsymbol{\epsilon}^{T}M^{T}|X\right]\\
&= ME\left[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^{T}|X\right]M^{T}\\
&= M\sigma_{\epsilon}^{2}I_{N}M^{T}\\
&= \sigma_{\epsilon}^{2}MM^{T}
\end{aligned}
$$

We have

The Covariance Matrix

$$
\begin{aligned}
E\left[\left(\widetilde{\boldsymbol{w}}-\boldsymbol{w}\right)\left(\widetilde{\boldsymbol{w}}-\boldsymbol{w}\right)^{T}|X\right] &= E\left[M\boldsymbol{\epsilon}\left(M\boldsymbol{\epsilon}\right)^{T}|X\right]\\
&= E\left[M\boldsymbol{\epsilon}\boldsymbol{\epsilon}^{T}M^{T}|X\right]\\
&= ME\left[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^{T}|X\right]M^{T}\\
&= M\sigma_{\epsilon}^{2}I_{N}M^{T}\\
&= \sigma_{\epsilon}^{2}MM^{T}
\end{aligned}
$$

Cinvestav

We have

The Covariance Matrix

$$
\begin{aligned}
E\left[\left(\widetilde{\boldsymbol{w}}-\boldsymbol{w}\right)\left(\widetilde{\boldsymbol{w}}-\boldsymbol{w}\right)^{T}|X\right] &= E\left[M\boldsymbol{\epsilon}\left(M\boldsymbol{\epsilon}\right)^{T}|X\right] \\
&= E\left[M\boldsymbol{\epsilon}\boldsymbol{\epsilon}^{T}M^{T}|X\right] \\
&= ME\left[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^{T}|X\right]M^{T} \\
&= M\sigma_{\epsilon}^{2}I_{N}M^{T} \\
&= \sigma_{\epsilon}^{2}MM^{T}
\end{aligned}
$$

We have

The Covariance Matrix

$$\begin{aligned} E\left[\left(\widetilde{\boldsymbol{w}}-\boldsymbol{w}\right)\left(\widetilde{\boldsymbol{w}}-\boldsymbol{w}\right)^{T}|X\right] &= E\left[M\boldsymbol{\epsilon}\left(M\boldsymbol{\epsilon}\right)^{T}|X\right] \\ &= E\left[M\boldsymbol{\epsilon}\boldsymbol{\epsilon}^{T}M^{T}|X\right] \\ &= ME\left[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^{T}|X\right]M^{T} \\ &= M\sigma_{\epsilon}^{2}I_{N}M^{T} \\ &= \sigma_{\epsilon}^{2}MM^{T} \end{aligned}$$

Finally

Given that $CX = 0$

$$MM^T = \left[\left(X^TX\right)^{-1}X^T + C\right]\left[\left(X^TX\right)^{-1}X^T + C\right]^T$$

$$= \left(X^TX\right)^{-1}X^TX\left(X^TX\right)^{-1} + \left(X^TX\right)^{-1}X^TC$$

$$+ CX\left(X^TX\right)^{-1} + CC^T$$

$$= \left(X^TX\right)^{-1} + CC^T$$

Finally

Given that $CX = 0$

$$
\begin{aligned}
MM^T &= \left[\left(X^TX\right)^{-1}X^T + C\right]\left[\left(X^TX\right)^{-1}X^T + C\right]^T \\
&= \left(X^TX\right)^{-1}X^TX\left(X^TX\right)^{-1} + \left(X^TX\right)^{-1}X^TC \\
&\quad + CX\left(X^TX\right)^{-1} + CC^T \\
&= \left(X^TX\right)^{-1} + CC^T
\end{aligned}
$$

Now the matrix CC^T is a $k \times k$ "cross products" matrix

- By construction is positive semi-definite

Cinvestav

Finally

Given that $CX = 0$

$$
\begin{aligned}
MM^T &= \left[\left(X^TX\right)^{-1}X^T + C\right]\left[\left(X^TX\right)^{-1}X^T + C\right]^T\\
&= \left(X^TX\right)^{-1}X^TX\left(X^TX\right)^{-1} + \left(X^TX\right)^{-1}X^TC\\
&+ CX\left(X^TX\right)^{-1} + CC^T\\
&= \left(X^TX\right)^{-1} + CC^T
\end{aligned}
$$

Now the matrix CC^T is a $k \times k$ "cross products" matrix

- By construction is positive semi-definite

Finally

Given that $CX = 0$

$$
\begin{aligned}
MM^T &= \left[\left(X^TX\right)^{-1}X^T + C\right]\left[\left(X^TX\right)^{-1}X^T + C\right]^T\\
&= \left(X^TX\right)^{-1}X^TX\left(X^TX\right)^{-1} + \left(X^TX\right)^{-1}X^TC\\
&+ CX\left(X^TX\right)^{-1} + CC^T\\
&= \left(X^TX\right)^{-1} + CC^T
\end{aligned}
$$

Now the matrix CC^T is a $k \times k$ "cross products" matrix

- By construction is positive semi-definite

Thus

Given

- The best estimator in a class of estimators is the one with the "smallest" covariance matrix

Where by "small"

- The covariance matrix associated with any other estimator in the class minus the covariance matrix of the best estimator is **a positive definite matrix**

Thus

Given

- The best estimator in a class of estimators is the one with the "smallest" covariance matrix

Where by "small"

- The covariance matrix associated with any other estimator in the class minus the covariance matrix of the best estimator is **a positive definite matrix**

Formally

The following difference is positive definite

$$MM^T + CC^T - Cov_{best}$$

Then

Since $MM^T + CC^T - Cov_{best}$ is minimized when we set the matrix C equal to the 0 matrix

- i.e. $M = \left(X^T X\right)^{-1} X$
  - The best estimator in the class $\widehat{\boldsymbol{w}}$.

Any other estimator M in this class

- It has strictly "larger" covariance matrix

Therefore the Least Square Error estimator $\widehat{w}$

- It is BLUE under the two conditions of mean independence and homoskedastic!!!

Then

Since $MM^T + CC^T - Cov_{best}$ is minimized when we set the matrix C equal to the 0 matrix

- i.e. $M = \left(X^T X\right)^{-1} X$
  - The best estimator in the class $\widehat{\boldsymbol{w}}$.

Any other estimator M in this class

- It has strictly "larger" covariance matrix

Therefore the Least Square Error estimator $\widehat{w}$

- It is BLUE under the two conditions of mean independence and homoskedastic!!!

Then

Since $MM^T + CC^T - Cov_{best}$ is minimized when we set the matrix C equal to the 0 matrix

- i.e. $M = \left(X^T X\right)^{-1} X$
  - The best estimator in the class $\widehat{\boldsymbol{w}}$.

Any other estimator M in this class

- It has strictly "larger" covariance matrix

Therefore the Least Square Error estimator $\widehat{\boldsymbol{w}}$

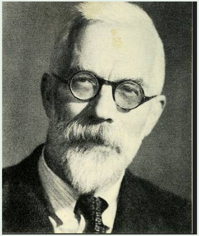
- It is BLUE under the two conditions of mean independence and homoskedastic!!!

Outline

Cinvestav

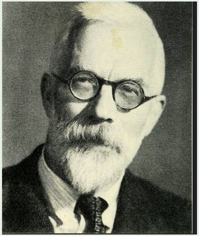
Invented Originally by

Sir Ronald Fisher



Invented Originally by

Sir Ronald Fisher

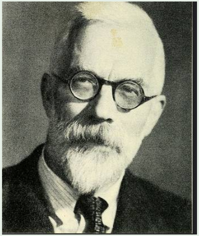


Anders Hald called him

"A genius who almost single-handedly created the foundations for modern statistical science."

Invented Originally by

Sir Ronald Fisher



Anders Hald called him

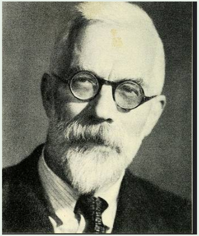
"A genius who almost single-handedly created the foundations for modern statistical science."

The Darkest Side

- In 1910 he joined the Eugenics Society at Cambridge, whose members included John Maynard Keynes, R. C. Punnett, and Horace Darwin.
- He opposed UNESCO's The Race Question, believing that evidence and everyday experience showed that human groups differ profoundly.

Invented Originally by

Sir Ronald Fisher



Anders Hald called him

"A genius who almost single-handedly created the foundations for modern statistical science."

The Darkest Side

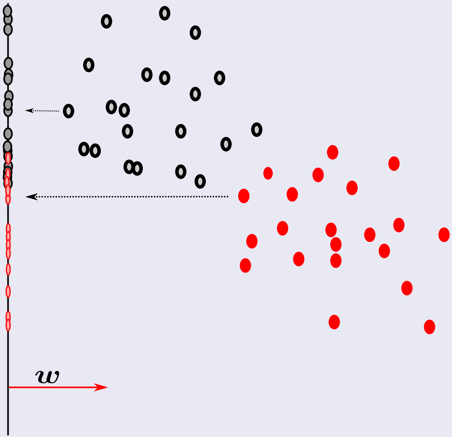
- In 1910 he joined the Eugenics Society at Cambridge, whose members included John Maynard Keynes, R. C. Punnett, and Horace Darwin.
- He opposed UNESCO's The Race Question, believing that evidence and everyday experience showed that human groups differ profoundly.

Outline

Cinvestav

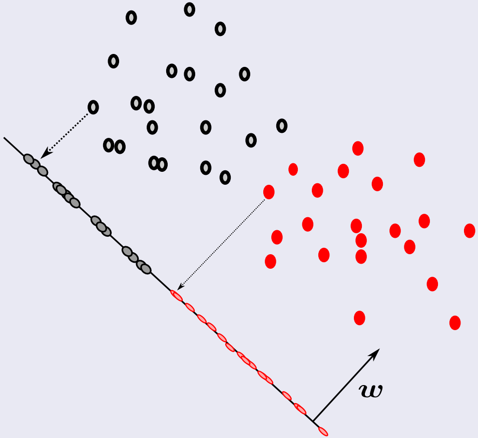
Intuition

Something Notable - Projecting into a Line



A Better Line

Something Notable - Projecting into a Line



Rotation

Projecting

Projecting well-separated samples onto an arbitrary line usually produces a confused mixture of samples from all of the classes and thus produces poor recognition performance.

Something Notable

However, moving and rotating the line around might result in an orientation for which the projected samples are well separated.

Fisher Linear Discriminant (FLD)

It is a discriminant analysis seeking directions that are efficient for discriminating binary classification problem.

Cinvestav

Rotation

Projecting

Projecting well-separated samples onto an arbitrary line usually produces a confused mixture of samples from all of the classes and thus produces poor recognition performance.

Something Notable

However, moving and rotating the line around might result in an orientation for which the projected samples are well separated.

Fisher Linear Discriminant (FLD)

It is a discriminant analysis seeking directions that are efficient for discriminating binary classification problem.

Rotation

Projecting

Projecting well-separated samples onto an arbitrary line usually produces a confused mixture of samples from all of the classes and thus produces poor recognition performance.

Something Notable

However, moving and rotating the line around might result in an orientation for which the projected samples are well separated.

Fisher Linear Discriminant (FLD)

It is a discriminant analysis seeking directions that are efficient for discriminating binary classification problem.

Outline

This is actually coming from...

Classifier as

A machine for dimensionality reduction.

This is actually coming from...

Classifier as

A machine for dimensionality reduction.

Initial Setup

We have:

- N d-dimensional samples $x_1, x_2, ..., x_N$.
- N_i is the number of samples in class C_i for i=1,2.

This is actually coming from...

Classifier as

A machine for dimensionality reduction.

Initial Setup

We have:

- N d-dimensional samples $x_1, x_2, ..., x_N$.
- N_i is the number of samples in class C_i for i=1,2.

Then, we ask for the projection of each x_i into the line by means of

$$y_i = w^T x_i \tag{9}$$

This is actually coming from...

Classifier as

A machine for dimensionality reduction.

Initial Setup

We have:

- N d-dimensional samples $x_1, x_2, ..., x_N$.
- N_i is the number of samples in class C_i for i=1,2.

Then, we ask for the projection of each x_i into the line by means of

$$y_i = w^T x_i \tag{9}$$

Cinvestav

This is actually coming from...

Classifier as

A machine for dimensionality reduction.

Initial Setup

We have:

- N d-dimensional samples $x_1, x_2, ..., x_N$.
- N_i is the number of samples in class C_i for i=1,2.

Then, we ask for the projection of each x_i into the line by means of

$$y_i = \boldsymbol{w}^T \boldsymbol{x}_i \tag{9}$$

Cinvestav

Outline

Cinvestav

Outline

Use the mean of each Class

Then

Select $\boldsymbol{w}$ such that class separation is maximized

Use the mean of each Class

Then

Select $\boldsymbol{w}$ such that class separation is maximized

We then define the mean sample for ecah class

1. $C_1 \Rightarrow \boldsymbol{m}_1 = \frac{1}{N_1} \sum_{i=1}^{N_1} \boldsymbol{x}_i$
2. $C_2 \Rightarrow \boldsymbol{m}_2 = \frac{1}{N_2} \sum_{i=1}^{N_2} \boldsymbol{x}_i$

Use the mean of each Class

Then

Select $\boldsymbol{w}$ such that class separation is maximized

We then define the mean sample for ecah class

1. $C_1 \Rightarrow \boldsymbol{m}_1 = \frac{1}{N_1}\sum_{i=1}^{N_1} \boldsymbol{x}_i$
2. $C_2 \Rightarrow \boldsymbol{m}_2 = \frac{1}{N_2}\sum_{i=1}^{N_2} \boldsymbol{x}_i$

Ok!!! This is giving us a measure of distance

Thus, we want to maximize the distance the projected means:

$$m_1 - m_2 = \boldsymbol{w}^T\left(\boldsymbol{m}_1 - \boldsymbol{m}_2\right) \quad (10)$$

where $m_k = \boldsymbol{w}^T\boldsymbol{m}_k$ for $k = 1, 2$.

Use the mean of each Class

Then

Select $\boldsymbol{w}$ such that class separation is maximized

We then define the mean sample for ecah class

1. $C_1 \Rightarrow \boldsymbol{m}_1 = \frac{1}{N_1} \sum_{i=1}^{N_1} \boldsymbol{x}_i$
2. $C_2 \Rightarrow \boldsymbol{m}_2 = \frac{1}{N_2} \sum_{i=1}^{N_2} \boldsymbol{x}_i$

Ok!!! This is giving us a measure of distance

Thus, we want to maximize the distance the projected means:

$$m_1 - m_2 = \boldsymbol{w}^T (\boldsymbol{m}_1 - \boldsymbol{m}_2) \quad (10)$$

where $m_k = \boldsymbol{w}^T \boldsymbol{m}_k$ for $k = 1, 2$.

Cinvestav

Use the mean of each Class

Then

Select $\boldsymbol{w}$ such that class separation is maximized

We then define the mean sample for ecah class

1. $C_1 \Rightarrow \boldsymbol{m}_1 = \frac{1}{N_1} \sum_{i=1}^{N_1} \boldsymbol{x}_i$
2. $C_2 \Rightarrow \boldsymbol{m}_2 = \frac{1}{N_2} \sum_{i=1}^{N_2} \boldsymbol{x}_i$

Ok!!! This is giving us a measure of distance

Thus, we want to maximize the distance the projected means:

$$m_1 - m_2 = \boldsymbol{w}^T \left(\boldsymbol{m}_1 - \boldsymbol{m}_2 \right) \tag{10}$$

where $m_k = \boldsymbol{w}^T \boldsymbol{m}_k$ for $k = 1, 2$.

However

We could simply seek

$$\max_{\boldsymbol{w}} \ \boldsymbol{w}^T (\boldsymbol{m}_1 - \boldsymbol{m}_2)$$
$$s.t. \sqrt{\boldsymbol{w}^T \boldsymbol{w}} = 1$$

After all

We do not care about the magnitude of w.

Cinvestav

However

We could simply seek

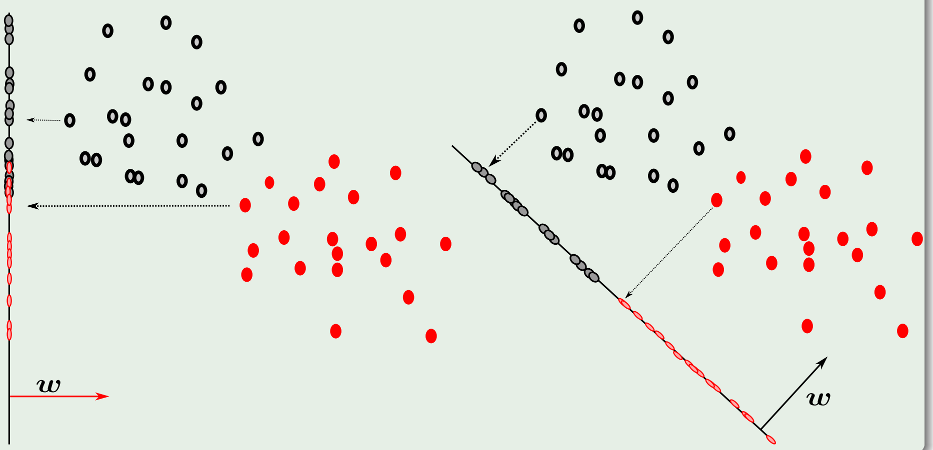
$$\max_{\boldsymbol{w}} \; \boldsymbol{w}^T (\boldsymbol{m}_1 - \boldsymbol{m}_2)$$
$$s.t. \sqrt{\boldsymbol{w}^T \boldsymbol{w}} = 1$$

After all

We do not care about the magnitude of $\boldsymbol{w}$.

Example

Here, we have the problem... The Scattering!!!



Outline

Cinvestav

Fixing the Problem

To obtain good separation of the projected data

The difference between the means should be large relative to some measure of the standard deviations for each class.

We define a SCATTER measure (Based in the Sample Variance)

$$s_k^2 = \sum_{x_i \in C_k} \left(w^T x_i - m_k\right)^2 = \sum_{y_i = w^T x_i \in C_k} \left(y_i - m_k\right)^2 \tag{11}$$

We define then within-class variance for the whole data

$$s_1^2 + s_2^2 \tag{12}$$

Fixing the Problem

To obtain good separation of the projected data

The difference between the means should be large relative to some measure of the standard deviations for each class.

We define a SCATTER measure (Based in the Sample Variance)

$$s_k^2 = \sum_{\boldsymbol{x}_i \in C_k} \left(\boldsymbol{w}^T \boldsymbol{x}_i - m_k \right)^2 = \sum_{y_i = \boldsymbol{w}^T \boldsymbol{x}_i \in C_k} (y_i - m_k)^2 \tag{11}$$

We define then within-class variance for the whole data

$$s_1^2 + s_2^2 \tag{12}$$

Fixing the Problem

To obtain good separation of the projected data

The difference between the means should be large relative to some measure of the standard deviations for each class.

We define a SCATTER measure (Based in the Sample Variance)

$$s_k^2 = \sum_{\boldsymbol{x}_i \in C_k} \left(\boldsymbol{w}^T \boldsymbol{x}_i - m_k\right)^2 = \sum_{y_i = \boldsymbol{w}^T \boldsymbol{x}_i \in C_k} (y_i - m_k)^2 \tag{11}$$

We define then within-class variance for the whole data

$$s_1^2 + s_2^2 \tag{12}$$

Cinvestav

Outline

Cinvestav

Finally, a Cost Function

The between-class variance

$$(m_1 - m_2)^2 \tag{13}$$

The Fisher criterion (A Ratio)

$$\frac{\text{between-class variance}}{\text{within-class variance}} \tag{14}$$

Finally

$$J(w) = \frac{(m_1 - m_2)^2}{s_1^2 + s_2^2} \tag{15}$$

Finally, a Cost Function

The between-class variance

$$(m_1 - m_2)^2 \tag{13}$$

The Fisher criterion (A Ratio)

$$\frac{\text{between-class variance}}{\text{within-class variance}} \tag{14}$$

Finally

$$J(w) = \frac{(m_1 - m_2)^2}{s_1^2 + s_2^2} \tag{15}$$

Finally, a Cost Function

The between-class variance

$$(m_1 - m_2)^2 \tag{13}$$

The Fisher criterion (A Ratio)

$$\frac{\text{between-class variance}}{\text{within-class variance}} \tag{14}$$

Finally

$$J(\boldsymbol{w}) = \frac{(m_1 - m_2)^2}{s_1^2 + s_2^2} \tag{15}$$

Cinvestav

Outline

We use a transformation to simplify our life

First

$$J(\boldsymbol{w}) = \frac{\left(\boldsymbol{w}^T \boldsymbol{m}_1 - \boldsymbol{w}^T \boldsymbol{m}_2\right)^2}{\sum_{y_i = \boldsymbol{w}^T \boldsymbol{x}_i \in C_1} (y_i - m_1)^2 + \sum_{y_i = \boldsymbol{w}^T \boldsymbol{x}_i \in C_2} (y_i - m_2)^2} \tag{16}$$

Second

$$\frac{\left(\boldsymbol{w}^T \boldsymbol{m}_1 - \boldsymbol{w}^T \boldsymbol{m}_2\right)\left(\boldsymbol{w}^T \boldsymbol{m}_1 - \boldsymbol{w}^T \boldsymbol{m}_2\right)^T}{\sum_{y_i = \boldsymbol{w}^T \boldsymbol{x}_i \in C_1} \left(\boldsymbol{w}^T \boldsymbol{x}_i - \boldsymbol{m}_1\right)\left(\boldsymbol{w}^T \boldsymbol{x}_i - \boldsymbol{m}_1\right)^T + \sum_{y_i = \boldsymbol{w}^T \boldsymbol{x}_i \in C_2} \left(\boldsymbol{w}^T \boldsymbol{x}_i - \boldsymbol{m}_2\right)\left(\boldsymbol{w}^T \boldsymbol{x}_i - \boldsymbol{m}_2\right)^T} \tag{17}$$

Third

$$\frac{\boldsymbol{w}^T (\boldsymbol{m}_1 - \boldsymbol{m}_2)\left(\boldsymbol{w}^T (\boldsymbol{m}_1 - \boldsymbol{m}_2)\right)^T}{\sum_{y_i = \boldsymbol{w}^T \boldsymbol{x}_i \in C_1} \boldsymbol{w}^T (\boldsymbol{x}_i - \boldsymbol{m}_1)\left(\boldsymbol{w}^T (\boldsymbol{x}_i - \boldsymbol{m}_1)\right)^T + \sum_{y_i = \boldsymbol{w}^T \boldsymbol{x}_i \in C_2} \boldsymbol{w}^T (\boldsymbol{x}_i - \boldsymbol{m}_2)\left(\boldsymbol{w}^T (\boldsymbol{x}_i - \boldsymbol{m}_2)\right)^T} \tag{18}$$

We use a transformation to simplify our life

First

$$J(\boldsymbol{w}) = \frac{\left(\boldsymbol{w}^T\boldsymbol{m_1} - \boldsymbol{w}^T\boldsymbol{m_2}\right)^2}{\sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i \in C_1}(y_i - m_1)^2 + \sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i \in C_2}(y_i - m_2)^2} \tag{16}$$

Second

$$\frac{\left(\boldsymbol{w}^T\boldsymbol{m_1} - \boldsymbol{w}^T\boldsymbol{m_2}\right)\left(\boldsymbol{w}^T\boldsymbol{m_1} - \boldsymbol{w}^T\boldsymbol{m_2}\right)^T}{\sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i \in C_1}\left(\boldsymbol{w}^T\boldsymbol{x}_i - m_1\right)\left(\boldsymbol{w}^T\boldsymbol{x}_i - m_1\right)^T + \sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i \in C_2}\left(\boldsymbol{w}^T\boldsymbol{x}_i - m_2\right)\left(\boldsymbol{w}^T\boldsymbol{x}_i - m_2\right)^T} \tag{17}$$

Third

$$\frac{\boldsymbol{w}^T(\boldsymbol{m_1} - \boldsymbol{m_2})\left(\boldsymbol{w}^T(\boldsymbol{m_1} - \boldsymbol{m_2})\right)^T}{\sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i \in C_1}\boldsymbol{w}^T(\boldsymbol{x}_i - \boldsymbol{m_1})\left(\boldsymbol{w}^T(\boldsymbol{x}_i - \boldsymbol{m_1})\right)^T + \sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i \in C_2}\boldsymbol{w}^T(\boldsymbol{x}_i - \boldsymbol{m_2})\left(\boldsymbol{w}^T(\boldsymbol{x}_i - \boldsymbol{m_2})\right)^T} \tag{18}$$

We use a transformation to simplify our life

First

$$J\left(\boldsymbol{w}\right)=\frac{\left(\boldsymbol{w}^{T}\boldsymbol{m}_{1}-\boldsymbol{w}^{T}\boldsymbol{m}_{2}\right)^{2}}{\sum_{y_{i}=\boldsymbol{w}^{T}\boldsymbol{x}_{i}\in C_{1}}\left(y_{i}-m_{1}\right)^{2}+\sum_{y_{i}=\boldsymbol{w}^{T}\boldsymbol{x}_{i}\in C_{2}}\left(y_{i}-m_{2}\right)^{2}} \tag{16}$$

Second

$$\frac{\left(\boldsymbol{w}^{T}\boldsymbol{m}_{1}-\boldsymbol{w}^{T}\boldsymbol{m}_{2}\right)\left(\boldsymbol{w}^{T}\boldsymbol{m}_{1}-\boldsymbol{w}^{T}\boldsymbol{m}_{2}\right)^{T}}{\sum_{y_{i}=\boldsymbol{w}^{T}\boldsymbol{x}_{i}\in C_{1}}\left(\boldsymbol{w}^{T}\boldsymbol{x}_{i}-m_{1}\right)\left(\boldsymbol{w}^{T}\boldsymbol{x}_{i}-m_{1}\right)^{T}+\sum_{y_{i}=\boldsymbol{w}^{T}\boldsymbol{x}_{i}\in C_{2}}\left(\boldsymbol{w}^{T}\boldsymbol{x}_{i}-m_{2}\right)\left(\boldsymbol{w}^{T}\boldsymbol{x}_{i}-m_{2}\right)^{T}} \tag{17}$$

Third

$$\frac{\boldsymbol{w}^{T}\left(\boldsymbol{m}_{1}-\boldsymbol{m}_{2}\right)\left(\boldsymbol{w}^{T}\left(\boldsymbol{m}_{1}-\boldsymbol{m}_{2}\right)\right)^{T}}{\sum_{y_{i}=\boldsymbol{w}^{T}\boldsymbol{x}_{i}\in C_{1}}\boldsymbol{w}^{T}\left(\boldsymbol{x}_{i}-\boldsymbol{m}_{1}\right)\left(\boldsymbol{w}^{T}\left(\boldsymbol{x}_{i}-\boldsymbol{m}_{1}\right)\right)^{T}+\sum_{y_{i}=\boldsymbol{w}^{T}\boldsymbol{x}_{i}\in C_{2}}\boldsymbol{w}^{T}\left(\boldsymbol{x}_{i}-\boldsymbol{m}_{2}\right)\left(\boldsymbol{w}^{T}\left(\boldsymbol{x}_{i}-\boldsymbol{m}_{2}\right)\right)^{T}} \tag{18}$$

Transformation

Fourth

$$\frac{\boldsymbol{w}^T\left(\boldsymbol{m}_1-\boldsymbol{m}_2\right)\left(\boldsymbol{m}_1-\boldsymbol{m}_2\right)^T\boldsymbol{w}}{\sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i\in C_1}\boldsymbol{w}^T\left(\boldsymbol{x}_i-\boldsymbol{m}_1\right)\left(\boldsymbol{x}_i-\boldsymbol{m}_1\right)^T\boldsymbol{w}+\sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i\in C_2}\boldsymbol{w}^T\left(\boldsymbol{x}_i-\boldsymbol{m}_2\right)\left(\boldsymbol{x}_i-\boldsymbol{m}_2\right)^T\boldsymbol{w}} \tag{19}$$

Fifth

$$\frac{\boldsymbol{w}^T\left(\boldsymbol{m}_1-\boldsymbol{m}_2\right)\left(\boldsymbol{m}_1-\boldsymbol{m}_2\right)^T\boldsymbol{w}}{\boldsymbol{w}^T\left[\sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i\in C_1}\left(\boldsymbol{x}_i-\boldsymbol{m}_1\right)\left(\boldsymbol{x}_i-\boldsymbol{m}_1\right)^T+\sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i\in C_2}\left(\boldsymbol{x}_i-\boldsymbol{m}_2\right)\left(\boldsymbol{x}_i-\boldsymbol{m}_2\right)^T\right]\boldsymbol{w}} \tag{20}$$

Now Rename

$$J(\boldsymbol{w})=\frac{\boldsymbol{w}^T S_B \boldsymbol{w}}{\boldsymbol{w}^T S_w \boldsymbol{w}} \tag{21}$$

Transformation

Fourth

$$\frac{\boldsymbol{w}^T\left(\boldsymbol{m}_1-\boldsymbol{m}_2\right)\left(\boldsymbol{m}_1-\boldsymbol{m}_2\right)^T\boldsymbol{w}}{\sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i\in C_1}\boldsymbol{w}^T\left(\boldsymbol{x}_i-\boldsymbol{m}_1\right)\left(\boldsymbol{x}_i-\boldsymbol{m}_1\right)^T\boldsymbol{w}+\sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i\in C_2}\boldsymbol{w}^T\left(\boldsymbol{x}_i-\boldsymbol{m}_2\right)\left(\boldsymbol{x}_i-\boldsymbol{m}_2\right)^T\boldsymbol{w}} \tag{19}$$

Fifth

$$\frac{\boldsymbol{w}^T\left(\boldsymbol{m}_1-\boldsymbol{m}_2\right)\left(\boldsymbol{m}_1-\boldsymbol{m}_2\right)^T\boldsymbol{w}}{\boldsymbol{w}^T\left[\sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i\in C_1}\left(\boldsymbol{x}_i-\boldsymbol{m}_1\right)\left(\boldsymbol{x}_i-\boldsymbol{m}_1\right)^T+\sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i\in C_2}\left(\boldsymbol{x}_i-\boldsymbol{m}_2\right)\left(\boldsymbol{x}_i-\boldsymbol{m}_2\right)^T\right]\boldsymbol{w}} \tag{20}$$

Now Rename

$$J(w)=\frac{w^TS_Bw}{w^TS_ww} \tag{21}$$

Transformation

Fourth

$$\frac{\boldsymbol{w}^T\left(\boldsymbol{m}_1-\boldsymbol{m}_2\right)\left(\boldsymbol{m}_1-\boldsymbol{m}_2\right)^T\boldsymbol{w}}{\sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i\in C_1}\boldsymbol{w}^T\left(\boldsymbol{x}_i-\boldsymbol{m}_1\right)\left(\boldsymbol{x}_i-\boldsymbol{m}_1\right)^T\boldsymbol{w}+\sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i\in C_2}\boldsymbol{w}^T\left(\boldsymbol{x}_i-\boldsymbol{m}_2\right)\left(\boldsymbol{x}_i-\boldsymbol{m}_2\right)^T\boldsymbol{w}} \tag{19}$$

Fifth

$$\frac{\boldsymbol{w}^T\left(\boldsymbol{m}_1-\boldsymbol{m}_2\right)\left(\boldsymbol{m}_1-\boldsymbol{m}_2\right)^T\boldsymbol{w}}{\boldsymbol{w}^T\left[\sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i\in C_1}\left(\boldsymbol{x}_i-\boldsymbol{m}_1\right)\left(\boldsymbol{x}_i-\boldsymbol{m}_1\right)^T+\sum_{y_i=\boldsymbol{w}^T\boldsymbol{x}_i\in C_2}\left(\boldsymbol{x}_i-\boldsymbol{m}_2\right)\left(\boldsymbol{x}_i-\boldsymbol{m}_2\right)^T\right]\boldsymbol{w}} \tag{20}$$

Now Rename

$$J\left(\boldsymbol{w}\right)=\frac{\boldsymbol{w}^T\boldsymbol{S}_B\boldsymbol{w}}{\boldsymbol{w}^T\boldsymbol{S}_w\boldsymbol{w}} \tag{21}$$

Cinvestav

Derive with respect to $\mathbf{w}$

Thus

$$\frac{dJ(\boldsymbol{w})}{d\boldsymbol{w}} = \frac{d\left(\boldsymbol{w}^T \boldsymbol{S}_B \boldsymbol{w}\right)\left(\boldsymbol{w}^T \boldsymbol{S}_w \boldsymbol{w}\right)^{-1}}{d\boldsymbol{w}} = 0 \tag{22}$$

Then

$$\frac{dJ(\boldsymbol{w})}{d\boldsymbol{w}} = \left(\boldsymbol{S}_B \boldsymbol{w} + \boldsymbol{S}_B^T \boldsymbol{w}\right)\left(\boldsymbol{w}^T \boldsymbol{S}_w \boldsymbol{w}\right)^{-1} - \left(\boldsymbol{w}^T \boldsymbol{S}_B \boldsymbol{w}\right)\left(\boldsymbol{w}^T \boldsymbol{S}_w \boldsymbol{w}\right)^{-2}\left(\boldsymbol{S}_w \boldsymbol{w} + \boldsymbol{S}_w^T \boldsymbol{w}\right) = 0 \tag{23}$$

Now because the symmetry in $\boldsymbol{S}_B$ and $\boldsymbol{S}_w$

$$\frac{dJ(\boldsymbol{w})}{d\boldsymbol{w}} = \frac{\boldsymbol{S}_B \boldsymbol{w}}{\left(\boldsymbol{w}^T \boldsymbol{S}_w \boldsymbol{w}\right)} - \frac{\boldsymbol{w}^T \boldsymbol{S}_B \boldsymbol{w} \boldsymbol{S}_w \boldsymbol{w}}{\left(\boldsymbol{w}^T \boldsymbol{S}_w \boldsymbol{w}\right)^2} = 0 \tag{24}$$

Derive with respect to $\mathbf{w}$

Thus

$$\frac{dJ\left(\boldsymbol{w}\right)}{d\boldsymbol{w}}=\frac{d\left(\boldsymbol{w}^{T}\boldsymbol{S}_{B}\boldsymbol{w}\right)\left(\boldsymbol{w}^{T}\boldsymbol{S}_{w}\boldsymbol{w}\right)^{-1}}{d\boldsymbol{w}}=0 \tag{22}$$

Then

$$\frac{dJ\left(\boldsymbol{w}\right)}{d\boldsymbol{w}}=\left(\boldsymbol{S}_{B}\boldsymbol{w}+\boldsymbol{S}_{B}^{T}w\right)\left(\boldsymbol{w}^{T}\boldsymbol{S}_{w}\boldsymbol{w}\right)^{-1}-\left(\boldsymbol{w}^{T}\boldsymbol{S}_{B}\boldsymbol{w}\right)\left(\boldsymbol{w}^{T}\boldsymbol{S}_{w}\boldsymbol{w}\right)^{-2}\left(\boldsymbol{S}_{w}\boldsymbol{w}+\boldsymbol{S}_{w}^{T}w\right)=0 \tag{23}$$

Now because the symmetry in S_B and S_w

$$\frac{dJ\left(\boldsymbol{w}\right)}{d\boldsymbol{w}}=\frac{\boldsymbol{S}_{B}\boldsymbol{w}}{\left(\boldsymbol{w}^{T}\boldsymbol{S}_{w}\boldsymbol{w}\right)}-\frac{\boldsymbol{w}^{T}\boldsymbol{S}_{B}\boldsymbol{w}\boldsymbol{S}_{w}\boldsymbol{w}}{\left(\boldsymbol{w}^{T}\boldsymbol{S}_{w}\boldsymbol{w}\right)^{2}}=0 \tag{24}$$

Derive with respect to $\mathbf{w}$

Thus

$$\frac{dJ\left(\boldsymbol{w}\right)}{d\boldsymbol{w}}=\frac{d\left(\boldsymbol{w}^{T}\boldsymbol{S}_{B}\boldsymbol{w}\right)\left(\boldsymbol{w}^{T}\boldsymbol{S}_{w}\boldsymbol{w}\right)^{-1}}{d\boldsymbol{w}}=0 \tag{22}$$

Then

$$\frac{dJ\left(\boldsymbol{w}\right)}{d\boldsymbol{w}}=\left(\boldsymbol{S}_{B}\boldsymbol{w}+\boldsymbol{S}_{B}^{T}w\right)\left(\boldsymbol{w}^{T}\boldsymbol{S}_{w}\boldsymbol{w}\right)^{-1}-\left(\boldsymbol{w}^{T}\boldsymbol{S}_{B}\boldsymbol{w}\right)\left(\boldsymbol{w}^{T}\boldsymbol{S}_{w}\boldsymbol{w}\right)^{-2}\left(\boldsymbol{S}_{w}\boldsymbol{w}+\boldsymbol{S}_{w}^{T}w\right)=0 \tag{23}$$

Now because the symmetry in $\boldsymbol{S}_B$ and $\boldsymbol{S}_w$

$$\frac{dJ\left(\boldsymbol{w}\right)}{d\boldsymbol{w}}=\frac{\boldsymbol{S}_{B}\boldsymbol{w}}{\left(\boldsymbol{w}^{T}\boldsymbol{S}_{w}\boldsymbol{w}\right)}-\frac{\boldsymbol{w}^{T}\boldsymbol{S}_{B}\boldsymbol{w}\boldsymbol{S}_{w}\boldsymbol{w}}{\left(\boldsymbol{w}^{T}\boldsymbol{S}_{w}\boldsymbol{w}\right)^{2}}=0 \tag{24}$$

Derive with respect to $\mathbf{w}$

Thus

$$\frac{dJ(\boldsymbol{w})}{d\boldsymbol{w}} = \frac{\boldsymbol{S}_B \boldsymbol{w}}{\left(\boldsymbol{w}^T \boldsymbol{S}_w \boldsymbol{w}\right)} - \frac{\boldsymbol{w}^T \boldsymbol{S}_B \boldsymbol{w} \boldsymbol{S}_w \boldsymbol{w}}{\left(\boldsymbol{w}^T \boldsymbol{S}_w \boldsymbol{w}\right)^2} = 0 \tag{25}$$

Then

$$\left(\boldsymbol{w}^T \boldsymbol{S}_w \boldsymbol{w}\right) \boldsymbol{S}_B \boldsymbol{w} = \left(\boldsymbol{w}^T \boldsymbol{S}_B \boldsymbol{w}\right) \boldsymbol{S}_w \boldsymbol{w} \tag{26}$$

Derive with respect to $\mathbf{w}$

Thus

$$\frac{dJ\left(\boldsymbol{w}\right)}{d\boldsymbol{w}} = \frac{\boldsymbol{S}_B\boldsymbol{w}}{\left(\boldsymbol{w}^T\boldsymbol{S}_w\boldsymbol{w}\right)} - \frac{\boldsymbol{w}^T\boldsymbol{S}_B\boldsymbol{w}\boldsymbol{S}_w\boldsymbol{w}}{\left(\boldsymbol{w}^T\boldsymbol{S}_w\boldsymbol{w}\right)^2} = 0 \tag{25}$$

Then

$$\left(\boldsymbol{w}^T\boldsymbol{S}_w\boldsymbol{w}\right)\boldsymbol{S}_B\boldsymbol{w} = \left(\boldsymbol{w}^T\boldsymbol{S}_B\boldsymbol{w}\right)\boldsymbol{S}_w\boldsymbol{w} \tag{26}$$

Now, Several Tricks!!!

First

$$S_B w = (m_1 - m_2)(m_1 - m_2)^T w = \alpha (m_1 - m_2) \tag{27}$$

Where $\alpha = (m_1 - m_2)^T w$ is a simple constant

It means that $S_B w$ is always in the direction $m_1 - m_2$!!!

In addition

$w^T S_w w$ and $w^T S_B w$ are constants

Now, Several Tricks!!!

First

$$S_B w = (m_1 - m_2)(m_1 - m_2)^T w = \alpha (m_1 - m_2) \tag{27}$$

Where $\alpha = (m_1 - m_2)^T w$ is a simple constant

It means that $S_B w$ is always in the direction $m_1 - m_2$!!!

In addition

$w^T S_w w$ and $w^T S_B w$ are constants

Now, Several Tricks!!!

First

$$\boldsymbol{S}_B\boldsymbol{w} = (\boldsymbol{m}_1 - \boldsymbol{m}_2)(\boldsymbol{m}_1 - \boldsymbol{m}_2)^T\boldsymbol{w} = \alpha(\boldsymbol{m}_1 - \boldsymbol{m}_2) \tag{27}$$

Where $\alpha = (\boldsymbol{m}_1 - \boldsymbol{m}_2)^T\boldsymbol{w}$ is a simple constant

It means that $\boldsymbol{S}_B\boldsymbol{w}$ is always in the direction $\boldsymbol{m}_1 - \boldsymbol{m}_2$!!!

In addition

$\boldsymbol{w}^T\boldsymbol{S}_w\boldsymbol{w}$ and $\boldsymbol{w}^T\boldsymbol{S}_B\boldsymbol{w}$ are constants

Cinvestav

Now, Several Tricks!!!

Finally, we only need the direction

$$\boldsymbol{S}_w \boldsymbol{w} \propto (\boldsymbol{m}_1 - \boldsymbol{m}_2) \Rightarrow \boldsymbol{w} \propto \boldsymbol{S}_w^{-1} (\boldsymbol{m}_1 - \boldsymbol{m}_2) \tag{28}$$

Now, Several Tricks!!!

Finally, we only need the direction

$$\boldsymbol{S}_w \boldsymbol{w} \propto (\boldsymbol{m}_1 - \boldsymbol{m}_2) \Rightarrow \boldsymbol{w} \propto \boldsymbol{S}_w^{-1} (\boldsymbol{m}_1 - \boldsymbol{m}_2) \tag{28}$$

Once the data is transformed into y_i

- Use a threshold $y_0 \Rightarrow x \in C_1$ iff $y(x) \geq y_0$ or $x \in C_2$ iff $y(x) < y_0$
- Or ML with a Gussian can be used to classify the new transformed data using a Naive Bayes (Central Limit Theorem and $y = w^T x$ sum of random variables).

Now, Several Tricks!!!

Finally, we only need the direction

$$S_w w \propto (m_1 - m_2) \Rightarrow w \propto S_w^{-1} (m_1 - m_2) \tag{28}$$

Once the data is transformed into y_i

- Use a threshold $y_0 \Rightarrow x \in C_1$ iff $y(x) \geq y_0$ or $x \in C_2$ iff $y(x) < y_0$
- Or ML with a Gussian can be used to classify the new transformed data using a Naive Bayes (Central Limit Theorem and $y = \boldsymbol{w}^T \boldsymbol{x}$ sum of random variables).

Outline

Cinvestav

Outline

Applications

Something Notable

- Bankruptcy prediction
  - In bankruptcy prediction based on accounting ratios and other financial variables, linear discriminant analysis was the first statistical method applied to systematically explain which firms entered bankruptcy vs. survived.
- Face recognition
  - In computerized face recognition, each face is represented by a large number of pixel values.
  - The linear combinations obtained using Fisher's linear discriminant are called Fisher faces.
- Marketing
  - In marketing, discriminant analysis was once often used to determine the factors which distinguish different types of customers and/or products on the basis of surveys or other forms of collected data.

Applications

Something Notable

- Bankruptcy prediction
  - In bankruptcy prediction based on accounting ratios and other financial variables, linear discriminant analysis was the first statistical method applied to systematically explain which firms entered bankruptcy vs. survived.
- Face recognition
  - In computerized face recognition, each face is represented by a large number of pixel values.
  - The linear combinations obtained using Fisher's linear discriminant are called Fisher faces.
- Marketing
  - In marketing, discriminant analysis was once often used to determine the factors which distinguish different types of customers and/or products on the basis of surveys or other forms of collected data.

Applications

Something Notable

- Bankruptcy prediction
  - In bankruptcy prediction based on accounting ratios and other financial variables, linear discriminant analysis was the first statistical method applied to systematically explain which firms entered bankruptcy vs. survived.
- Face recognition
  - In computerized face recognition, each face is represented by a large number of pixel values.
  - The linear combinations obtained using Fisher's linear discriminant are called Fisher faces.
- Marketing
  - In marketing, discriminant analysis was once often used to determine the factors which distinguish different types of customers and/or products on the basis of surveys or other forms of collected data.

Applications

Something Notable

- Bankruptcy prediction
  - In bankruptcy prediction based on accounting ratios and other financial variables, linear discriminant analysis was the first statistical method applied to systematically explain which firms entered bankruptcy vs. survived.
- Face recognition
  - In computerized face recognition, each face is represented by a large number of pixel values.
  - The linear combinations obtained using Fisher's linear discriminant are called Fisher faces.
- Marketing
  - In marketing, discriminant analysis was once often used to determine the factors which distinguish different types of customers and/or products on the basis of surveys or other forms of collected data.

Applications

Something Notable

- Bankruptcy prediction
  - In bankruptcy prediction based on accounting ratios and other financial variables, linear discriminant analysis was the first statistical method applied to systematically explain which firms entered bankruptcy vs. survived.
- Face recognition
  - In computerized face recognition, each face is represented by a large number of pixel values.
  - The linear combinations obtained using Fisher's linear discriminant are called Fisher faces.
- Marketing
  - In marketing, discriminant analysis was once often used to determine the factors which distinguish different types of customers and/or products on the basis of surveys or other forms of collected data.

Applications

Something Notable

- Bankruptcy prediction
  - In bankruptcy prediction based on accounting ratios and other financial variables, linear discriminant analysis was the first statistical method applied to systematically explain which firms entered bankruptcy vs. survived.
- Face recognition
  - In computerized face recognition, each face is represented by a large number of pixel values.
  - The linear combinations obtained using Fisher's linear discriminant are called Fisher faces.
- Marketing
  - In marketing, discriminant analysis was once often used to determine the factors which distinguish different types of customers and/or products on the basis of surveys or other forms of collected data.

Applications

Something Notable

- Bankruptcy prediction
  - In bankruptcy prediction based on accounting ratios and other financial variables, linear discriminant analysis was the first statistical method applied to systematically explain which firms entered bankruptcy vs. survived.
- Face recognition
  - In computerized face recognition, each face is represented by a large number of pixel values.
  - The linear combinations obtained using Fisher's linear discriminant are called Fisher faces.
- Marketing
  - In marketing, discriminant analysis was once often used to determine the factors which distinguish different types of customers and/or products on the basis of surveys or other forms of collected data.

Applications

Something Notable

- Bankruptcy prediction
  - In bankruptcy prediction based on accounting ratios and other financial variables, linear discriminant analysis was the first statistical method applied to systematically explain which firms entered bankruptcy vs. survived.
- Face recognition
  - In computerized face recognition, each face is represented by a large number of pixel values.
  - The linear combinations obtained using Fisher's linear discriminant are called Fisher faces.
- Marketing
  - In marketing, discriminant analysis was once often used to determine the factors which distinguish different types of customers and/or products on the basis of surveys or other forms of collected data.

Applications

Something Notable

- Biomedical studies
    - The main application of discriminant analysis in medicine is the assessment of severity state of a patient and prognosis of disease outcome.

Applications

Something Notable

- Biomedical studies
  - The main application of discriminant analysis in medicine is the assessment of severity state of a patient and prognosis of disease outcome.

Please

Your Reading Material, it is about the Multi-class

4.1.6 Fisher's discriminant for multiple classes AT "Pattern Recognition" by Bishop

Outline

Outline

Cinvestav

Relation with Least Squared Error

First

The least-squares approach to the determination of a linear discriminant was based on the goal of making the model predictions **as close as possible** to a set of target values.

Second

The Fisher criterion was derived by requiring maximum class separation in the output space.

Relation with Least Squared Error

First

The least-squares approach to the determination of a linear discriminant was based on the goal of making the model predictions **as close as possible** to a set of target values.

Second

The Fisher criterion was derived by requiring maximum class separation in the output space.

How do we do this?

First

- We have N samples.
- We have N_1 samples for class C_1.
- We have N_2 samples for class C_2.

How do we do this?

First

- We have N samples.
- We have N_1samples for class C_1.
- We have N_2 samples for class C_2.

So, we decide the following for the targets on each class

- We have then for class C_1 is $t_1 = \frac{N}{N_1}$.
- We have then for class C_2 is $t_2 = -\frac{N}{N_2}$.

Cinvestav

How do we do this?

First

- We have N samples.
- We have N_1samples for class C_1.
- We have N_2 samples for class C_2.

So, we decide the following for the targets on each class

- We have then for class C_1 is $t_1 = \frac{N}{N_1}$.
- We have then for class C_2 is $t_2 = -\frac{N}{N_2}$.

How do we do this?

First

- We have N samples.
- We have N_1samples for class C_1.
- We have N_2 samples for class C_2.

So, we decide the following for the targets on each class

- We have then for class C_1 is $t_1 = \frac{N}{N_1}$.
- We have then for class C_2 is $t_2 = -\frac{N}{N_2}$.

Cinvestav

Thus

The new cost function (Our Classic Linear Model)

$$E = \frac{1}{2}\sum_{n=1}^{N}\left(\boldsymbol{w}^T\boldsymbol{x}_n + w_0 - t_n\right)^2 \tag{29}$$

Deriving with respect to $\boldsymbol{w}$

$$\sum_{n=1}^{N}\left(\boldsymbol{w}^T\boldsymbol{x}_n + w_0 - t_n\right)\boldsymbol{x}_n = 0 \tag{30}$$

Deriving with respect to w_0

$$\sum_{n=1}^{N}\left(\boldsymbol{w}^T\boldsymbol{x}_n + w_0 - t_n\right) = 0 \tag{31}$$

Cinvestav

Thus

The new cost function (Our Classic Linear Model)

$$E = \frac{1}{2}\sum_{n=1}^{N}\left(\boldsymbol{w}^T\boldsymbol{x}_n + w_0 - t_n\right)^2 \tag{29}$$

Deriving with respect to $\boldsymbol{w}$

$$\sum_{n=1}^{N}\left(\boldsymbol{w}^T\boldsymbol{x}_n + w_0 - t_n\right)\boldsymbol{x}_n = 0 \tag{30}$$

Deriving with respect to w_0

$$\sum_{n=1}^{N}\left(\boldsymbol{w}^T\boldsymbol{x}_n + w_0 - t_n\right) = 0 \tag{31}$$

Cinvestav

Thus

The new cost function (Our Classic Linear Model)

$$E = \frac{1}{2} \sum_{n=1}^{N} \left(\boldsymbol{w}^T \boldsymbol{x}_n + w_0 - t_n \right)^2 \tag{29}$$

Deriving with respect to $\boldsymbol{w}$

$$\sum_{n=1}^{N} \left(\boldsymbol{w}^T \boldsymbol{x}_n + w_0 - t_n \right) \boldsymbol{x}_n = 0 \tag{30}$$

Deriving with respect to w_0

$$\sum_{n=1}^{N} \left(\boldsymbol{w}^T \boldsymbol{x}_n + w_0 - t_n \right) = 0 \tag{31}$$

Cinvestav

Then

We have that

$$\sum_{n=1}^{N}\left(\boldsymbol{w}^T\boldsymbol{x}_n + w_0 - t_n\right) = \sum_{n=1}^{N}\left(\boldsymbol{w}^T\boldsymbol{x}_n + w_0\right) - \sum_{n=1}^{N} t_n$$

$$= \sum_{n=1}^{N}\left(\boldsymbol{w}^T\boldsymbol{x}_n + w_0\right) - N_1\frac{N}{N_1} + N_2\frac{N}{N_2}$$

$$= \sum_{n=1}^{N}\left(\boldsymbol{w}^T\boldsymbol{x}_n + w_0\right)$$

Then

We have that

$$
\begin{aligned}
\sum_{n=1}^{N}\left(\boldsymbol{w}^T\boldsymbol{x}_n + w_0 - t_n\right) &= \sum_{n=1}^{N}\left(\boldsymbol{w}^T\boldsymbol{x}_n + w_0\right) - \sum_{n=1}^{N} t_n \\
&= \sum_{n=1}^{N}\left(\boldsymbol{w}^T\boldsymbol{x}_n + w_0\right) - N_1\frac{N}{N_1} + N_2\frac{N}{N_2} \\
&= \sum_{n=1}^{N}\left(\boldsymbol{w}^T\boldsymbol{x}_n + w_0\right)
\end{aligned}
$$

Then

$$
\left(\sum_{n=1}^{N}\boldsymbol{w}^T\boldsymbol{x}_n\right) + N w_0 = 0
$$

Then

We have that

$$\begin{aligned}\sum_{n=1}^{N}\left(\boldsymbol{w}^{T}\boldsymbol{x}_{n}+w_{0}-t_{n}\right) &= \sum_{n=1}^{N}\left(\boldsymbol{w}^{T}\boldsymbol{x}_{n}+w_{0}\right)-\sum_{n=1}^{N}t_{n}\\ &= \sum_{n=1}^{N}\left(\boldsymbol{w}^{T}\boldsymbol{x}_{n}+w_{0}\right)-N_{1}\frac{N}{N_{1}}+N_{2}\frac{N}{N_{2}}\\ &= \sum_{n=1}^{N}\left(\boldsymbol{w}^{T}\boldsymbol{x}_{n}+w_{0}\right)\end{aligned}$$

Then

$$\left(\sum_{n=1}^{N}\boldsymbol{w}^{T}\boldsymbol{x}_{n}\right)+Nw_{0}=0$$

Then

We have that

$$w_0 = -\boldsymbol{w}^T \left(\frac{1}{N} \sum_{n=1}^{N} \boldsymbol{x}_n \right)$$

We rename $\frac{1}{N} \sum_{n=1}^{N} \boldsymbol{x}_n = \boldsymbol{m}$

$$\boldsymbol{m} = \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{x}_n = \frac{1}{N} \left[N_1 \boldsymbol{m}_1 + N_2 \boldsymbol{m}_2 \right]$$

Finally

$$w_0 = -\boldsymbol{w}^T \boldsymbol{m}$$

Then

We have that

$$w_0 = -\boldsymbol{w}^T \left(\frac{1}{N} \sum_{n=1}^{N} \boldsymbol{x}_n \right)$$

We rename $\frac{1}{N} \sum_{n=1}^{N} \boldsymbol{x}_n = \boldsymbol{m}$

$$\boldsymbol{m} = \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{x}_n = \frac{1}{N} \left[N_1 \boldsymbol{m}_1 + N_2 \boldsymbol{m}_2 \right]$$

Finally

$$w_0 = -\boldsymbol{w}^T \boldsymbol{m}$$

Then

We have that

$$w_0 = -\boldsymbol{w}^T \left(\frac{1}{N} \sum_{n=1}^{N} \boldsymbol{x}_n \right)$$

We rename $\frac{1}{N} \sum_{n=1}^{N} \boldsymbol{x}_n = \boldsymbol{m}$

$$\boldsymbol{m} = \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{x}_n = \frac{1}{N} \left[N_1 \boldsymbol{m}_1 + N_2 \boldsymbol{m}_2 \right]$$

Finally

$$w_0 = -\boldsymbol{w}^T \boldsymbol{m}$$

Cinvestav

Now

In a similar way

$$\sum_{n=1}^{N}\left(\boldsymbol{w}^{T}\boldsymbol{x}_n + w_0\right)\boldsymbol{x}_n - \sum_{n=1}^{N} t_n \boldsymbol{x}_n = 0$$

Thus, we have

Something Notable

$$\sum_{n=1}^{N}\left(\boldsymbol{w}^{T}\boldsymbol{x}_{n}+w_{0}\right)\boldsymbol{x}_{n}-\frac{N}{N_{1}}\sum_{n=1}^{N_{1}}\boldsymbol{x}_{n}+\frac{N}{N_{2}}\sum_{n=1}^{N_{2}}\boldsymbol{x}_{n}=0$$

Thus

$$\sum_{n=1}^{N}\left(\boldsymbol{w}^{T}\boldsymbol{x}_{n}+w_{0}\right)\boldsymbol{x}_{n}-N\left(\boldsymbol{m}_{1}-\boldsymbol{m}_{2}\right)=0$$

Thus, we have

Something Notable

$$\sum_{n=1}^{N}\left(\boldsymbol{w}^{T}\boldsymbol{x}_{n}+w_{0}\right)\boldsymbol{x}_{n}-\frac{N}{N_{1}}\sum_{n=1}^{N_{1}}\boldsymbol{x}_{n}+\frac{N}{N_{2}}\sum_{n=1}^{N_{2}}\boldsymbol{x}_{n}=0$$

Thus

$$\sum_{n=1}^{N}\left(\boldsymbol{w}^{T}\boldsymbol{x}_{n}+w_{0}\right)\boldsymbol{x}_{n}-N\left(\boldsymbol{m}_{1}-\boldsymbol{m}_{2}\right)=0$$

Next

Then, using $w_0 = -\boldsymbol{w}^T\boldsymbol{m}$

$$\sum_{n=1}^{N}\left(\boldsymbol{w}^T\boldsymbol{x}_n - \boldsymbol{w}^T\boldsymbol{m}\right)\boldsymbol{x}_n = N\left(\boldsymbol{m}_1 - \boldsymbol{m}_2\right)$$

Thus

$$\left[\sum_{n=1}^{N}\left(\boldsymbol{w}^T\boldsymbol{x}_n - \boldsymbol{w}^T\boldsymbol{m}\right)\boldsymbol{x}_n\right] = N\left(\boldsymbol{m}_1 - \boldsymbol{m}_2\right)$$

Next

Then, using $w_0 = -\boldsymbol{w}^T \boldsymbol{m}$

$$\sum_{n=1}^{N} \left(\boldsymbol{w}^T \boldsymbol{x}_n - \boldsymbol{w}^T \boldsymbol{m} \right) \boldsymbol{x}_n = N \left(\boldsymbol{m}_1 - \boldsymbol{m}_2 \right)$$

Thus

$$\left[\sum_{n=1}^{N} \left(\boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{x}_n - \boldsymbol{w}^T \boldsymbol{m} \right) \boldsymbol{x}_n \right] = N \left(\boldsymbol{m}_1 - \boldsymbol{m}_2 \right)$$

Now, Do you have the solution?

You have a version in Duda and Hart Section 5.8

$$\widehat{\boldsymbol{w}} = \left(\boldsymbol{X}^T \boldsymbol{X}\right)^{-1} \boldsymbol{X}^T \boldsymbol{y}$$

Thus

$$\boldsymbol{X}^T \boldsymbol{X} \widehat{\boldsymbol{w}} = \boldsymbol{X}^T \boldsymbol{y}$$

Now, we rewrite the data matrix

$$\boldsymbol{X} = \begin{bmatrix} \mathbf{1}_1 & \boldsymbol{X}_1 \\ -\mathbf{1}_2 & -\boldsymbol{X}_2 \end{bmatrix}$$

Cinvestav

Now, Do you have the solution?

You have a version in Duda and Hart Section 5.8

$$\widehat{\boldsymbol{w}} = \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

Thus

$$\boldsymbol{X}^T\boldsymbol{X}\widehat{\boldsymbol{w}} = \boldsymbol{X}^T\boldsymbol{y}$$

Now, we rewrite the data matrix

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{1}_1 & \boldsymbol{X}_1 \\ -\boldsymbol{1}_2 & -\boldsymbol{X}_2 \end{bmatrix}$$

Cinvestav

Now, Do you have the solution?

You have a version in Duda and Hart Section 5.8

$$\widehat{\boldsymbol{w}} = \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

Thus

$$\boldsymbol{X}^T\boldsymbol{X}\widehat{\boldsymbol{w}} = \boldsymbol{X}^T\boldsymbol{y}$$

Now, we rewrite the data matrix

$$\boldsymbol{X} = \begin{bmatrix} \mathbf{1}_1 & \boldsymbol{X}_1 \\ -\mathbf{1}_2 & -\boldsymbol{X}_2 \end{bmatrix}$$

Cinvestav

In addition

Our old augmented $\boldsymbol{w}$

$$\boldsymbol{w} = \begin{bmatrix} w_0 \\ \boldsymbol{w} \end{bmatrix}$$

And our new y

$$y = \begin{bmatrix} \frac{N}{N_1}\mathbf{1}_1 \\ \frac{N}{N_2}\mathbf{1}_2 \end{bmatrix} \tag{32}$$

In addition

Our old augmented $\boldsymbol{w}$

$$\boldsymbol{w} = \begin{bmatrix} w_0 \\ \boldsymbol{w} \end{bmatrix}$$

And our new $\boldsymbol{y}$

$$\boldsymbol{y} = \begin{bmatrix} \frac{N}{N_1}\mathbf{1}_1 \\ \frac{N}{N_2}\mathbf{1}_2 \end{bmatrix} \tag{32}$$

Cinvestav

Thus, we have

Something Notable

$$\begin{bmatrix} \mathbf{1}_1^T & -\mathbf{1}_2^T \\ \boldsymbol{X}_1^T & -\boldsymbol{X}_2^T \end{bmatrix} \begin{bmatrix} \mathbf{1}_1 & \boldsymbol{X}_1 \\ -\mathbf{1}_2 & -\boldsymbol{X}_2 \end{bmatrix} \begin{bmatrix} w_0 \\ \boldsymbol{w} \end{bmatrix} = \begin{bmatrix} \mathbf{1}_1^T & -\mathbf{1}_2^T \\ \boldsymbol{X}_1^T & -\boldsymbol{X}_2^T \end{bmatrix} \begin{bmatrix} \frac{N}{N_1}\mathbf{1}_1 \\ \frac{N}{N_2}\mathbf{1}_2 \end{bmatrix}$$

Thus, we have

Something Notable

$$\begin{bmatrix} \mathbf{1}_1^T & -\mathbf{1}_2^T \\ \boldsymbol{X}_1^T & -\boldsymbol{X}_2^T \end{bmatrix} \begin{bmatrix} \mathbf{1}_1 & \boldsymbol{X}_1 \\ -\mathbf{1}_2 & -\boldsymbol{X}_2 \end{bmatrix} \begin{bmatrix} w_0 \\ \boldsymbol{w} \end{bmatrix} = \begin{bmatrix} \mathbf{1}_1^T & -\mathbf{1}_2^T \\ \boldsymbol{X}_1^T & -\boldsymbol{X}_2^T \end{bmatrix} \begin{bmatrix} \frac{N}{N_1}\mathbf{1}_1 \\ \frac{N}{N_2}\mathbf{1}_2 \end{bmatrix}$$

Thus, if we use the following definitions for $i = 1, 2$

- $\boldsymbol{m}_i = \frac{1}{N_i} \sum_{\boldsymbol{x} \in C_i} \boldsymbol{x}$
- $S_w = \sum_{\boldsymbol{x}_i \in C_1} (\boldsymbol{x}_i - \boldsymbol{m}_1)(\boldsymbol{x}_i - \boldsymbol{m}_1)^T + \sum_{\boldsymbol{x}_i \in C_2} (\boldsymbol{x}_i - \boldsymbol{m}_2)(\boldsymbol{x}_i - \boldsymbol{m}_2)^T$

Thus, we have

Something Notable

$$\begin{bmatrix} \mathbf{1}_1^T & -\mathbf{1}_2^T \\ \boldsymbol{X}_1^T & -\boldsymbol{X}_2^T \end{bmatrix} \begin{bmatrix} \mathbf{1}_1 & \boldsymbol{X}_1 \\ -\mathbf{1}_2 & -\boldsymbol{X}_2 \end{bmatrix} \begin{bmatrix} w_0 \\ \boldsymbol{w} \end{bmatrix} = \begin{bmatrix} \mathbf{1}_1^T & -\mathbf{1}_2^T \\ \boldsymbol{X}_1^T & -\boldsymbol{X}_2^T \end{bmatrix} \begin{bmatrix} \frac{N}{N_1}\mathbf{1}_1 \\ \frac{N}{N_2}\mathbf{1}_2 \end{bmatrix}$$

Thus, if we use the following definitions for $i = 1, 2$

- $\boldsymbol{m}_i = \frac{1}{N_i} \sum_{\boldsymbol{x} \in C_i} \boldsymbol{x}$
- $S_w = \sum_{\boldsymbol{x}_i \in C_1} (\boldsymbol{x}_i - \boldsymbol{m}_1)(\boldsymbol{x}_i - \boldsymbol{m}_1)^T + \sum_{\boldsymbol{x}_i \in C_2} (\boldsymbol{x}_i - \boldsymbol{m}_2)(\boldsymbol{x}_i - \boldsymbol{m}_2)^T$

Cinvestav

Then

If we multiply the previous matrices

$$\begin{bmatrix} N & \left(N_1 \boldsymbol{m}_1 + N_2 \boldsymbol{m}_2\right)^T \\ \left(N_1 \boldsymbol{m}_1 + N_2 \boldsymbol{m}_2\right) & S_w + N_1 \boldsymbol{m}_1 \boldsymbol{m}_1^T + N_2 \boldsymbol{m}_2 \boldsymbol{m}_2^T \end{bmatrix} \begin{bmatrix} w_0 \\ \boldsymbol{w} \end{bmatrix} = \begin{bmatrix} 0 \\ N \left[\boldsymbol{m}_1 - \boldsymbol{m}_2\right] \end{bmatrix}$$

Then

$$\begin{bmatrix} N w_0 + \left(N_1 \boldsymbol{m}_1 + N_2 \boldsymbol{m}_2\right)^T \boldsymbol{w} \\ \left(N_1 \boldsymbol{m}_1 + N_2 \boldsymbol{m}_2\right) w_0 + \left[S_w + N_1 \boldsymbol{m}_1 \boldsymbol{m}_1^T + N_2 \boldsymbol{m}_2 \boldsymbol{m}_2^T\right] \end{bmatrix} = \begin{bmatrix} 0 \\ N \left[\boldsymbol{m}_1 - \boldsymbol{m}_2\right] \end{bmatrix}$$

Then

If we multiply the previous matrices

$$\begin{bmatrix} N & (N_1\boldsymbol{m}_1 + N_2\boldsymbol{m}_2)^T \\ (N_1\boldsymbol{m}_1 + N_2\boldsymbol{m}_2) & S_w + N_1\boldsymbol{m}_1\boldsymbol{m}_1^T + N_2\boldsymbol{m}_2\boldsymbol{m}_2^T \end{bmatrix} \begin{bmatrix} w_0 \\ \boldsymbol{w} \end{bmatrix} = \begin{bmatrix} 0 \\ N\left[\boldsymbol{m}_1 - \boldsymbol{m}_2\right] \end{bmatrix}$$

Then

$$\begin{bmatrix} Nw_0 + (N_1\boldsymbol{m}_1 + N_2\boldsymbol{m}_2)^T\boldsymbol{w} \\ (N_1\boldsymbol{m}_1 + N_2\boldsymbol{m}_2)\,w_0 + \left[S_w + N_1\boldsymbol{m}_1\boldsymbol{m}_1^T + N_2\boldsymbol{m}_2\boldsymbol{m}_2^T\right] \end{bmatrix} = \begin{bmatrix} 0 \\ N\left[\boldsymbol{m}_1 - \boldsymbol{m}_2\right] \end{bmatrix}$$

Cinvestav

Thus

We have that

$$w_0 = -\boldsymbol{w}^T \boldsymbol{m}$$

$$\left[\frac{1}{N} S_w + \frac{N_1 N_2}{N^2} (\boldsymbol{m}_1 - \boldsymbol{m}_2)(\boldsymbol{m}_1 - \boldsymbol{m}_2)^T\right] \boldsymbol{w} = \boldsymbol{m}_1 - \boldsymbol{m}_2$$

Thus

Since the vector $(\boldsymbol{m}_1 - \boldsymbol{m}_2)(\boldsymbol{m}_1 - \boldsymbol{m}_2)^T \boldsymbol{w}$ is in the direction of $\boldsymbol{m}_1 - \boldsymbol{m}_2$

$$\alpha = \frac{N_1 N_2}{N^2} (\boldsymbol{m}_1 - \boldsymbol{m}_2)^T \boldsymbol{w}$$

We have that

$$\frac{1}{N} S_w \boldsymbol{w} = (1 - \alpha)(\boldsymbol{m}_1 - \boldsymbol{m}_2)$$

Thus

We have that

$$w_0 = -\boldsymbol{w}^T \boldsymbol{m}$$

$$\left[\frac{1}{N} S_w + \frac{N_1 N_2}{N^2} (\boldsymbol{m}_1 - \boldsymbol{m}_2)(\boldsymbol{m}_1 - \boldsymbol{m}_2)^T\right] \boldsymbol{w} = \boldsymbol{m}_1 - \boldsymbol{m}_2$$

Thus

Since the vector $(\boldsymbol{m}_1 - \boldsymbol{m}_2)(\boldsymbol{m}_1 - \boldsymbol{m}_2)^T \boldsymbol{w}$ is in the direction of $\boldsymbol{m}_1 - \boldsymbol{m}_2$

$$\alpha = \frac{N_1 N_2}{N^2} (\boldsymbol{m}_1 - \boldsymbol{m}_2)^T \boldsymbol{w}$$

We have that

$$\frac{1}{N} S_w \boldsymbol{w} = (1 - \alpha)(\boldsymbol{m}_1 - \boldsymbol{m}_2)$$

Thus

We have that

$$
\begin{aligned}
w_0 &= -\boldsymbol{w}^T \boldsymbol{m} \\
\left[\frac{1}{N} S_w + \frac{N_1 N_2}{N^2} \left(\boldsymbol{m}_1 - \boldsymbol{m}_2\right) \left(\boldsymbol{m}_1 - \boldsymbol{m}_2\right)^T\right] \boldsymbol{w} &= \boldsymbol{m}_1 - \boldsymbol{m}_2
\end{aligned}
$$

Thus

Since the vector $\left(\boldsymbol{m}_1 - \boldsymbol{m}_2\right) \left(\boldsymbol{m}_1 - \boldsymbol{m}_2\right)^T \boldsymbol{w}$ is in the direction of $\boldsymbol{m}_1 - \boldsymbol{m}_2$

$$
\alpha = \frac{N_1 N_2}{N^2} \left(\boldsymbol{m}_1 - \boldsymbol{m}_2\right)^T \boldsymbol{w}
$$

We have that

$$
\frac{1}{N} S_w \boldsymbol{w} = (1 - \alpha) \left(\boldsymbol{m}_1 - \boldsymbol{m}_2\right)
$$

Finally

We have that

$$\boldsymbol{w} = (1-\alpha)\, N S_w^{-1} (\boldsymbol{m}_1 - \boldsymbol{m}_2) \propto S_w^{-1} (\boldsymbol{m}_1 - \boldsymbol{m}_2) \tag{33}$$

Cinvestav

Outline

Exercises

Machine Learning Theodoridis

Chapter 7

- 7.10, 7.13

Bishop

Chapter 4

- 4.4, 4.5, 4.6, 4.8

Exercises

Machine Learning Theodoridis

Chapter 7

- 7.10, 7.13

Bishop

Chapter 4

- 4.4, 4.5, 4.6, 4.8