

Machine Learning

Textbook: Machine Learning Refined, by Watt, Borhani, and Katsaggelos, 2nd edition:
MLRefined.com

Previous Lecture

- ♦ Many machine-learning objectives are finite sums: $g = (1/P) \sum_p g_p$.
- ♦ Full-batch gradient descent uses every observation before each update; SGD uses one; mini-batch methods use b observations.
- ♦ A uniformly sampled stochastic gradient is unbiased: $E[\nabla g_I] = \nabla g$.
- ♦ Mini-batches average several stochastic gradients, generally reducing gradient variance while retaining frequent updates.
- ♦ One epoch is one pass through the data; shuffled epochs prevent systematic ordering from dominating batches.
- ♦ With noisy gradients, learning-rate choice and decay affect the tradeoff between rapid progress and late-stage stability.



§8.2 - Orthogonal Projection and Data

So far:

supervised learning used inputs
and known outputs.

Now:

suppose we have only data vectors
 $\mathbf{x}_1, \dots, \mathbf{x}_P$.

No response variable.

Can we discover lower-dimensional structure
inside the data themselves?

We begin with orthogonal projection.

Unsupervised Data

Suppose each observation has

N

features:

$$\mathbf{x}_p \in \mathbb{R}^N.$$

We have no labels

y_p .

Possible goals:

find lower-dimensional structure;

compress the observations;

visualize high-dimensional data;

discover groups or patterns.

This is **unsupervised learning**.

Our first tool:

projection onto a subspace.

Project a Vector Onto a Line

Let

$$\mathbf{u} \in \mathbb{R}^N$$

be a unit vector:

$$\|\mathbf{u}\| = 1.$$

The line through the origin in direction

$$\mathbf{u}$$

is

$$\text{span}\{\mathbf{u}\}.$$

Given

$$\mathbf{x},$$

we seek the point on this line

closest to

$$\mathbf{x}.$$

Write that point as

$$c\mathbf{u}.$$

What value of c is best?

Projection as a Least-Squares Problem

Minimize squared distance:

$$h(c) = \|\mathbf{x} - c\mathbf{u}\|^2.$$

Expand:

$$\begin{aligned} h(c) &= (\mathbf{x} - c\mathbf{u})^T (\mathbf{x} - c\mathbf{u}) \\ &= \|\mathbf{x}\|^2 - 2c\mathbf{u}^T \mathbf{x} + c^2 \|\mathbf{u}\|^2. \end{aligned}$$

Since

$$\|\mathbf{u}\| = 1:$$

$$h(c) = \|\mathbf{x}\|^2 - 2c\mathbf{u}^T \mathbf{x} + c^2.$$

Differentiate:

$$h'(c) = -2\mathbf{u}^T \mathbf{x} + 2c.$$

Thus

$$c^* = \mathbf{u}^T \mathbf{x}.$$

Projection Onto a Unit Direction

The scalar coordinate is

$$z = \mathbf{u}^T \mathbf{x}.$$

The projected vector is

$$\hat{\mathbf{x}} = (\mathbf{u}^T \mathbf{x}) \mathbf{u}.$$

Equivalently:

$$\hat{\mathbf{x}} = \mathbf{u} \mathbf{u}^T \mathbf{x}.$$

Interpretation:

encode the vector by the one number

$$z = \mathbf{u}^T \mathbf{x};$$

reconstruct it by

$$\hat{\mathbf{x}} = z \mathbf{u}.$$

We have replaced an N -dimensional vector with one coordinate.

A Two-Dimensional Example

Let

$$\mathbf{x} = \begin{bmatrix} 3 \\ 1 \end{bmatrix},$$

and

$$\mathbf{u} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

Coordinate:

$$\begin{aligned} z &= \mathbf{u}^T \mathbf{x} \\ &= \frac{3+1}{\sqrt{2}} \\ &= 2\sqrt{2}. \end{aligned}$$

Projection:

$$\begin{aligned} \hat{\mathbf{x}} &= 2\sqrt{2} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} 2 \\ 2 \end{bmatrix}. \end{aligned}$$

The Projection Residual

Define the reconstruction residual

$$\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}}.$$

For the example:

$$\begin{aligned} \mathbf{e} &= \begin{bmatrix} 3 \\ 1 \end{bmatrix} - \begin{bmatrix} 2 \\ 2 \end{bmatrix} \\ &= \begin{bmatrix} 1 \\ -1 \end{bmatrix}. \end{aligned}$$

Check orthogonality:

$$\begin{aligned} &\mathbf{u}^T \mathbf{e} \\ &= \frac{1-1}{\sqrt{2}} \\ &= 0. \end{aligned}$$

The shortest residual is perpendicular to the projection line.

Why the Residual Is Orthogonal

Since

$$\hat{\mathbf{x}} = (\mathbf{u}^T \mathbf{x}) \mathbf{u},$$

we have

$$\begin{aligned} &\mathbf{u}^T \mathbf{e} \\ &= \mathbf{u}^T [\mathbf{x} - (\mathbf{u}^T \mathbf{x}) \mathbf{u}] \\ &= \mathbf{u}^T \mathbf{x} - (\mathbf{u}^T \mathbf{x}) \mathbf{u}^T \mathbf{u} \\ &= \mathbf{u}^T \mathbf{x} - \mathbf{u}^T \mathbf{x} \\ &= 0. \end{aligned}$$

This orthogonality condition
characterizes the least-squares projection.

Pythagorean Decomposition

We have

$$\mathbf{x} = \hat{\mathbf{x}} + \mathbf{e},$$

with

$$\hat{\mathbf{x}} \perp \mathbf{e}.$$

Therefore

$$\|\mathbf{x}\|^2 = \|\hat{\mathbf{x}}\|^2 + \|\mathbf{e}\|^2.$$

For a unit direction:

$$\|\hat{\mathbf{x}}\|^2 = (\mathbf{u}^T \mathbf{x})^2.$$

So

$$\|\mathbf{x}\|^2 = (\mathbf{u}^T \mathbf{x})^2 + \|\mathbf{x} - (\mathbf{u}^T \mathbf{x}) \mathbf{u}\|^2.$$

Energy captured by the projection
plus reconstruction error

equals total squared length.

What if the Direction Is Not Unit Length?

Let

$$\mathbf{v} \neq \mathbf{0}.$$

Project onto

$$\text{span}\{\mathbf{v}\}.$$

The optimal coefficient is

$$c^* = \frac{\mathbf{v}^T \mathbf{x}}{\mathbf{v}^T \mathbf{v}}.$$

Therefore

$$\text{proj}_{\mathbf{v}} \mathbf{x} = \frac{\mathbf{v}^T \mathbf{x}}{\mathbf{v}^T \mathbf{v}} \mathbf{v}.$$

Normalizing the direction first

gives the same projected vector.

Unit vectors simplify the formulas.

Projection Matrix Onto a Line

For unit

$$\mathbf{u},$$

define

$$P = \mathbf{u} \mathbf{u}^T.$$

Then

$$\hat{\mathbf{x}} = P \mathbf{x}.$$

For

$$\mathbf{u} = \frac{1}{\sqrt{2}}(1, 1)^T:$$

$$P = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}.$$

Apply to

$$\mathbf{x} = (3, 1)^T:$$

$$\begin{aligned} P \mathbf{x} &= \frac{1}{2} \begin{bmatrix} 4 \\ 4 \end{bmatrix} \\ &= (2, 2)^T. \end{aligned}$$

Two Properties of an Orthogonal Projection Matrix

First:

$$\begin{aligned} P^T &= (\mathbf{u} \mathbf{u}^T)^T \end{aligned}$$

$$\begin{aligned}
 &= \mathbf{u}\mathbf{u}^T \\
 &= P.
 \end{aligned}$$

Thus

$$P^T = P.$$

Second:

$$\begin{aligned}
 &P^2 \\
 &= \mathbf{u}\mathbf{u}^T\mathbf{u}\mathbf{u}^T \\
 &= \mathbf{u}(\mathbf{u}^T\mathbf{u})\mathbf{u}^T \\
 &= P.
 \end{aligned}$$

Thus

$$P^2 = P.$$

Projecting twice does nothing new.

Projection Onto a Higher-Dimensional Subspace

Now let

$$\mathbf{u}_1, \dots, \mathbf{u}_d$$

be orthonormal vectors.

Put them in the columns of

$$U = \begin{bmatrix} | & & | \\ \mathbf{u}_1 & \cdots & \mathbf{u}_d \\ | & & | \end{bmatrix} \in \mathbb{R}^{N \times d}.$$

Orthonormality means

$$U^T U = I_d.$$

The columns span a

$$d\text{-dimensional subspace of } \mathbb{R}^N.$$

Usually:

$$d < N.$$

Coordinates in the Subspace

For

$$\mathbf{x} \in \mathbb{R}^N,$$

the lower-dimensional coordinate vector is

$$\mathbf{z} = U^T \mathbf{x} \in \mathbb{R}^d.$$

Its entries are

$$z_j = \mathbf{u}_j^T \mathbf{x}.$$

Reconstruct:

$$\hat{\mathbf{x}} = U\mathbf{z} = UU^T\mathbf{x}.$$

So:

encode

$$\mathbf{x} \mapsto \mathbf{z} = U^T\mathbf{x};$$

decode

$$\mathbf{z} \mapsto \hat{\mathbf{x}} = U\mathbf{z}.$$

This is linear dimensionality reduction.

Projection Matrix Onto a Subspace

For orthonormal columns:

$$P_U = UU^T.$$

Then

$$\hat{\mathbf{x}} = P_U\mathbf{x}.$$

Again:

$$P_U^T = P_U,$$

and

$$P_U^2 = P_U.$$

The residual

$$\mathbf{e} = \mathbf{x} - UU^T\mathbf{x}$$

satisfies

$$U^T\mathbf{e} = \mathbf{0}.$$

Thus the error is orthogonal

to every direction in the retained subspace.

Projection Onto the Orthogonal Complement

Let

$$P = UU^T.$$

Then

$$I - P$$

projects onto the orthogonal complement

of the column space of U .

Indeed:

$$\hat{\mathbf{x}} = P\mathbf{x}$$

is the retained part,

while

$$\mathbf{e} = (I - P)\mathbf{x}$$

is the discarded part.

The decomposition is

$$\mathbf{x} = P\mathbf{x} + (I - P)\mathbf{x},$$

with the two pieces orthogonal.

A Three-Dimensional Example

Let

$$U = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}.$$

Its columns span the

x_1x_2 -plane.

For

$$\mathbf{x} = \begin{bmatrix} 2 \\ -1 \\ 5 \end{bmatrix}:$$

coordinates:

$$\begin{aligned} \mathbf{z} &= U^T \mathbf{x} \\ &= \begin{bmatrix} 2 \\ -1 \end{bmatrix}. \end{aligned}$$

Reconstruction:

$$\begin{aligned} \hat{\mathbf{x}} &= U\mathbf{z} \\ &= \begin{bmatrix} 2 \\ -1 \\ 0 \end{bmatrix}. \end{aligned}$$

Residual:

$$\mathbf{e} = \begin{bmatrix} 0 \\ 0 \\ 5 \end{bmatrix}.$$

What if the Basis Is Not Orthonormal?

Suppose the columns of

$$A \in \mathbb{R}^{N \times d}$$

are linearly independent,

but not necessarily orthonormal.

Find coefficients

$$\mathbf{z}$$

by minimizing
 $\|A\mathbf{z} - \mathbf{x}\|^2$.

Normal equations:
 $A^T A \mathbf{z} = A^T \mathbf{x}$.

Formally:
 $\mathbf{z}^* = (A^T A)^{-1} A^T \mathbf{x}$.

Then
 $\hat{\mathbf{x}} = A(A^T A)^{-1} A^T \mathbf{x}$.

Numerically:
solve the system;
do not explicitly form the inverse.

This Is the Same Geometry as Least Squares Regression

Recall Lesson 1.10.

We minimized
 $\|X\mathbf{w} - \mathbf{y}\|^2$.

The fitted vector
 $X\mathbf{w}^*$
was
the orthogonal projection of
 $\mathbf{y}$
onto
 $\text{Col}(X)$.

Now:
we minimize
 $\|A\mathbf{z} - \mathbf{x}\|^2$.

The reconstructed vector
 $A\mathbf{z}^*$
is the orthogonal projection of
 $\mathbf{x}$
onto
 $\text{Col}(A)$.

Same mathematics.

Different machine-learning purpose.

From One Vector to a Dataset

Store observations in the rows of
 X :

$$X = \begin{bmatrix} ---\mathbf{x}_1^T--- \\ ---\mathbf{x}_2^T--- \\ \vdots \\ ---\mathbf{x}_P^T--- \end{bmatrix} \in \mathbb{R}^{P \times N}.$$

Rows:
observations.

Columns:
features.

This is the same row convention
we used in supervised learning.

Before projecting a data cloud,
we usually center it.

The Mean Vector

Define the featurewise mean:

$$\boldsymbol{\mu} = \frac{1}{P} \sum_{p=1}^P \mathbf{x}_p.$$

Centered observation:

$$\mathbf{q}_p = \mathbf{x}_p - \boldsymbol{\mu}.$$

The centered observations satisfy

$$\frac{1}{P} \sum_{p=1}^P \mathbf{q}_p = \mathbf{0}.$$

Centering moves the center of the data cloud
to the origin.

That makes projection through the origin
geometrically meaningful for the data.

Why Center?

A linear subspace
always passes through the origin.

But a data cloud usually does not.

If we center first:
project

$$\mathbf{q} = \mathbf{x} - \boldsymbol{\mu}$$

onto a subspace through the origin.

Then return to the original coordinates:

$$\hat{\mathbf{x}} = \boldsymbol{\mu} + UU^T(\mathbf{x} - \boldsymbol{\mu}).$$

Thus in the original coordinate system,

we are projecting onto an **affine subspace**
passing through the mean.

A Small Dataset

Consider four observations:

p	x_1	x_2
1	0	1
2	1	2
3	2	2
4	3	3

Mean:

$$\boldsymbol{\mu} = \begin{bmatrix} 1.5 \\ 2 \end{bmatrix}.$$

Centered observations:

- $(-1.5, -1)$
- $(-0.5, 0)$
- $(0.5, 0)$
- $(1.5, 1)$

Their average is
 $(0, 0)$.

Project the Centered Data Onto a Diagonal Line

Choose

$$\mathbf{u} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

For each centered observation:

$$z_p = \mathbf{u}^T \mathbf{q}_p.$$

Coordinates:

$$\mathbf{z} = \left(-\frac{5}{2\sqrt{2}}, -\frac{1}{2\sqrt{2}}, \frac{1}{2\sqrt{2}}, \frac{5}{2\sqrt{2}} \right).$$

Each two-dimensional point
is now represented by one scalar coordinate.

Reconstruct From One Coordinate

For each observation:

$$\hat{\mathbf{q}}_p = z_p \mathbf{u}.$$

The reconstructed centered points are

- $(-1.25, -1.25)$
- $(-0.25, -0.25)$
- $(0.25, 0.25)$
- $(1.25, 1.25)$

Return to the original coordinates:

$$\hat{\mathbf{x}}_p = \boldsymbol{\mu} + \hat{\mathbf{q}}_p.$$

Thus

- $(0.25, 0.75)$
- $(1.25, 1.75)$
- $(1.75, 2.25)$
- $(2.75, 3.25)$

All reconstructions lie on one line through the data mean.

Reconstruction Residuals

Centered residuals:

$$\mathbf{e}_p = \mathbf{q}_p - \hat{\mathbf{q}}_p.$$

They are

- $(-0.25, 0.25)$
- $(-0.25, 0.25)$
- $(0.25, -0.25)$
- $(0.25, -0.25)$

Each residual is perpendicular to

$$\mathbf{u}.$$

Each squared reconstruction error is

$$\begin{aligned} &0.25^2 + 0.25^2 \\ &= 0.125. \end{aligned}$$

Total squared reconstruction error:

$$0.5.$$

Mean squared reconstruction distance:

$$0.125.$$

Compare Another Projection Direction

Instead choose the

x_1 -axis:

$$\mathbf{u} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}.$$

Then the second centered coordinate is discarded.

Squared errors are
1, 0, 0, 1.

Total squared reconstruction error:
2.

Mean:
0.5.

The diagonal direction gave only
0.125
mean squared reconstruction distance.

So:
the choice of projection direction matters.

Which direction is best?

That is the PCA question.

Data Matrix Form

Let X_c
contain centered observations in its rows:

$$X_c = \begin{bmatrix} \text{---} \mathbf{q}_1^T \text{---} \\ \vdots \\ \text{---} \mathbf{q}_P^T \text{---} \end{bmatrix}.$$

For an orthonormal basis
 $U \in \mathbb{R}^{N \times d}$:

low-dimensional coordinates:
 $Z = X_c U \in \mathbb{R}^{P \times d}$.

Reconstructed centered data:
 $\hat{X}_c = ZU^T = X_c U U^T$.

Reconstructed original data:
 $\hat{X} = \hat{X}_c + \boldsymbol{\mu}$
where the mean vector is added to every row.

Reconstruction Error for the Whole Dataset

For centered observations:
 $\mathbf{e}_p = \mathbf{q}_p - U U^T \mathbf{q}_p$.

A natural total error is

$$E(U) = \sum_{p=1}^P \|\mathbf{q}_p - UU^T \mathbf{q}_p\|^2.$$

Matrix form:

$$E(U) = \|X_c - X_c UU^T\|_F^2.$$

Here

$$\|\cdot\|_F$$

is the Frobenius norm:

the square root of the sum of squares of all matrix entries.

This error measures information lost
by the lower-dimensional representation.

Captured Energy Versus Reconstruction Error

For each centered observation:

$$\begin{aligned} &\|\mathbf{q}_p\|^2 \\ &= \|UU^T \mathbf{q}_p\|^2 \\ &+ \|\mathbf{q}_p - UU^T \mathbf{q}_p\|^2. \end{aligned}$$

Summing over observations:

$$\sum_p \|\mathbf{q}_p\|^2 = \sum_p \|U^T \mathbf{q}_p\|^2 + E(U).$$

The left side is fixed
once the data are given.

Therefore:

minimizing reconstruction error
is equivalent to
maximizing projected squared length.

This equivalence is the bridge to PCA.

For a One-Dimensional Subspace

Let

$$d = 1$$

and

$$U = \mathbf{u}$$

with

$$\|\mathbf{u}\| = 1.$$

Then captured energy is

$$\sum_{p=1}^P (\mathbf{u}^T \mathbf{q}_p)^2.$$

Reconstruction error is

$$\sum_{p=1}^P \|\mathbf{q}_p - (\mathbf{u}^T \mathbf{q}_p) \mathbf{u}\|^2.$$

Next lesson:

choose the unit vector

$\mathbf{u}$

that optimizes these equivalent criteria.

That vector will be the first principal component direction.

Projection in Python

Project one vector:

```
import numpy as np

x = np.array([3., 1.])

u = np.array([1., 1.])
u = u / np.linalg.norm(u)

z = u @ x
x_hat = z * u

residual = x - x_hat
```

New Python idea:

`np`

computes vector length.

The expression

$\mathbf{u} @ \mathbf{x}$

computes the dot product.

Projection Matrix in Python

Construct

```
 $P = \mathbf{u}\mathbf{u}^T$ :
P = np.outer(u, u)

x_hat = P @ x
```

New Python idea:

`np(u,u)`

forms the outer product

$\mathbf{u}\mathbf{u}^T$.

Check the projection-matrix identities:

```
np.allclose(P.T, P)
np.allclose(P @ P, P)
```

Both should return

`True`.

Center Data in Python

```
X = np.array([[0., 1.],
               [1., 2.],
               [2., 2.]])
```

```
[3., 3.]])
```

```
mu = np.mean(X, axis=0)
```

```
Xc = X - mu
```

Here

axis=0

takes the mean down the rows,
producing one mean for each feature.

Check:

```
np.mean(Xc, axis=0)
```

The result is approximately
(0,0).

Project the Entire Dataset

Use the diagonal direction:

```
u = np.array([1., 1.])  
u = u / np.linalg.norm(u)
```

```
U = u.reshape(-1, 1)
```

```
Z = Xc @ U  
Xc_hat = Z @ U.T
```

```
X_hat = Xc_hat + mu
```

Dimensions:

X_c is
 $P \times N$;

U is
 $N \times 1$;

Z is
 $P \times 1$.

We have compressed each observation
from two numbers to one.

Compute Reconstruction Error in Python

```
errors = Xc - Xc_hat
```

```
squared_errors = np.sum(  
    errors**2,  
    axis=1  
)
```

```
mean_reconstruction_error = np.mean(  
    squared_errors  
)
```

For the diagonal direction:
mean_reconstruction_error = 0.125.

For
 $u = (1, 0)^T$:
mean_reconstruction_error = 0.5.

Next:
let the data determine the best direction.

What Projection Does — and Does Not — Do

Projection can:

- reduce dimension;
- compress observations;
- remove components orthogonal to a chosen subspace;
- create coordinates for visualization.

But projection alone does not tell us
which subspace to choose.

So far:
we supplied
 U .

Principal component analysis will choose
 U
from the data.

What did we learn?

- ◆ Orthogonal projection is the least-squares approximation of a vector by a point in a chosen subspace.
- ◆ For a unit direction $\mathbf{u}$, the coordinate is $z = \mathbf{u}^T \mathbf{x}$ and the reconstruction is $\hat{\mathbf{x}} = \mathbf{u} \mathbf{u}^T \mathbf{x}$.
- ◆ For an orthonormal basis U , coordinates are $\mathbf{z} = U^T \mathbf{x}$ and the projection matrix is UU^T .
- ◆ Projection residuals are orthogonal to the retained subspace,
giving a Pythagorean decomposition of squared length.
- ◆ Centering lets a linear subspace through the origin represent an affine subspace through the data mean.
- ◆ Minimizing reconstruction error is equivalent to maximizing the squared length captured by the projection.



Next

- Projection compresses data once a subspace has been chosen.
- Different directions produce different reconstruction errors.
- We now want the direction that preserves as much of the centered data as possible.
- **1.23 - Principal Component Analysis**