PMAI LAB

Mathematics • Python • Artificial Intelligence

LINEAR ALGEBRA FOR MACHINE LEARNING

Lecture 06

Least-Squares Problems, Normal Equations, Linear Regression, and Model Fitting

Lecture focus Finding the best approximate solution when data is noisy or inconsistent, interpreting regression geometrically, and fitting models reliably with NumPy.

Dr. Hira Benish

Associate Professor of Mathematics

Student Learning Handout | Includes NumPy practice and review exercises

1 Lecture Overview

Real data rarely satisfies a mathematical model exactly. Measurements contain noise, observations may conflict, and there are often more equations than unknown parameters. Least squares replaces the demand for an exact solution with the search for the best approximate solution.

This lecture connects projections with least-squares estimation, derives the normal equations, develops simple linear regression, introduces residual metrics, and explains why modern software uses QR- or SVD-based solvers.

Learning Outcomes

After completing this lecture, students should be able to:

- explain why overdetermined systems are usually inconsistent;
- define residuals and the least-squares objective;
- interpret a least-squares solution as an orthogonal projection;
- derive and apply the normal equations;
- formulate linear and polynomial regression using a design matrix;
- calculate SSE, MSE, RMSE, and R^2;
- explain rank deficiency and numerical conditioning;
- fit and evaluate a model using `np.linalg.lstsq`; and
- apply regression to a sensor-calibration problem.

1.1 From Exact Systems to Data Fitting

Exact system	Least-squares model
$A\mathbf{x} = \mathbf{b}$	$A\hat{\mathbf{x}} \approx \mathbf{b}$
Requires consistency	Allows noisy or conflicting observations
Zero residual	Smallest possible residual norm
Often square	Commonly overdetermined: more rows than columns
Solves equations exactly	Fits parameters to data

2 The Least-Squares Problem

Consider

$$A\mathbf{x} \approx \mathbf{b}, \qquad A \in \mathbb{R}^{m\times n}, \quad m > n.$$

There are more observations than unknown parameters. An exact solution may not exist because **b** may not lie in the column space of A.

2.1 Residual Vector

For any proposed parameter vector **x**, define the residual

$$\mathbf{r} = \mathbf{b} - A\mathbf{x}.$$

Each entry is

$$r_i = \text{observed value} - \text{predicted value}.$$

The least-squares estimate is

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \|\mathbf{b} - A\mathbf{x}\|_2^2.$$

Since

$$\|\mathbf{r}\|_2^2 = r_1^2 + \cdots + r_m^2,$$

least squares minimizes the sum of squared residuals.

2.2 Why Squares?

- Positive and negative residuals do not cancel.
- Large errors receive greater penalty.
- The objective is differentiable and mathematically convenient.
- Under Gaussian error assumptions, least squares is also a maximum-likelihood estimator.

Least Squares Is an Approximation Method

The fitted vector $A\hat{\mathbf{x}}$ is not generally equal to **b**. It is the vector in Col(A) that is closest to **b** in Euclidean distance.

Machine-Learning Connection

Training a linear regression model means choosing weights that minimize a loss. Ordinary least squares uses squared residual error as that loss function.

3 Projection and the Normal Equations

The best prediction

$$\widehat{\mathbf{b}} = A\widehat{\mathbf{x}}$$

is the orthogonal projection of **b** onto Col(A). Therefore, the residual

$$\widehat{\mathbf{r}} = \mathbf{b} - A\widehat{\mathbf{x}}$$

must be orthogonal to every column of A.
This gives

$$A^T\widehat{\mathbf{r}} = \mathbf{0}.$$

Substituting the residual produces

$$A^T(\mathbf{b} - A\widehat{\mathbf{x}}) = \mathbf{0},$$

and hence the **normal equations**

$$A^T A\widehat{\mathbf{x}} = A^T\mathbf{b}.$$

If the columns of A are linearly independent, then A^TA is invertible and

$$\widehat{\mathbf{x}} = (A^TA)^{-1}A^T\mathbf{b}.$$

3.1 Projection Matrix

The fitted vector can be written as

$$\widehat{\mathbf{b}} = P\mathbf{b}, \qquad P = A(A^TA)^{-1}A^T.$$

The residual is $(I - P)\mathbf{b}$.

Geometric Optimality

For every other $\mathbf{y} \in$ Col(A),

$$\|\mathbf{b} - \widehat{\mathbf{b}}\|_2 \le \|\mathbf{b} - \mathbf{y}\|_2.$$

The least-squares prediction is the unique closest point in the column space, even when the parameter vector is not unique.

The formula with $(A^TA)^{-1}$ is useful for theory. In software, use a least-squares solver rather than forming the inverse explicitly.

4 Simple Linear Regression

For observations (x_i, y_i), a fitted line has the form

$$\widehat{y}_i = \beta_0 + \beta_1 x_i.$$

The design matrix and parameter vector are

$$A = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_m \end{bmatrix}, \qquad \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}.$$

The first column of ones represents the intercept.

4.1 Worked Example

Consider the data

$$(1, 2), \qquad (2, 2.5), \qquad (3, 4).$$

Then

$$A = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{bmatrix}, \qquad \mathbf{y} = \begin{bmatrix} 2 \\ 2.5 \\ 4 \end{bmatrix}.$$

The least-squares line is

$$\widehat{y} = 0.8333 + 1.0000x.$$

The fitted values are approximately

$$\widehat{\mathbf{y}} = \begin{bmatrix} 1.8333 \\ 2.8333 \\ 3.8333 \end{bmatrix},$$

and the residuals are

$$\mathbf{r} = \mathbf{y} - \widehat{\mathbf{y}} = \begin{bmatrix} 0.1667 \\ -0.3333 \\ 0.1667 \end{bmatrix}.$$

Notice that the residuals sum to zero because the model contains an intercept.

Machine-Learning Connection

The intercept shifts the prediction line, while the slope describes the expected change in the target for a one-unit change in the input. Interpretation depends on units, feature scaling, and the validity of the model assumptions.

5 Residual Metrics and Model Forms

Let y_i be observed values and $\widehat{y}_i$ be predictions.

5.1 Common Regression Metrics

$$
\begin{aligned}
\text{SSE} &= \sum_{i=1}^{m}(y_i - \widehat{y}_i)^2, \\
\text{MSE} &= \frac{1}{m}\sum_{i=1}^{m}(y_i - \widehat{y}_i)^2, \\
\text{RMSE} &= \sqrt{\text{MSE}}, \\
R^2 &= 1 - \frac{\sum_i (y_i - \widehat{y}_i)^2}{\sum_i (y_i - \overline{y})^2}.
\end{aligned}
$$

RMSE has the same units as the target. R^2 compares the model with predicting the mean target; it can be negative on unseen data when the model performs worse than that baseline.

5.2 Multiple Linear Regression

With n features,

$$\widehat{\mathbf{y}} = A\boldsymbol{\beta},$$

where each row of A contains the features of one observation, usually with a leading 1 for the intercept.

5.3 Polynomial Regression

A quadratic model

$$\widehat{y} = \beta_0 + \beta_1 x + \beta_2 x^2$$

is still linear in its parameters. Its design matrix uses columns 1, x, and x^2.

Residual Analysis

A good model should not leave a systematic pattern in its residuals. Curvature, changing spread, or clusters in residuals may indicate missing nonlinear structure, unequal variance, or different data groups.

Low training error does not guarantee good prediction on new data. Model assessment should use validation or test data, especially when comparing models of different complexity.

6 Numerical Stability and Regularization

The normal equations are mathematically correct, but computing $A^T A$ can worsen numerical conditioning. Roughly,

$$\kappa(A^T A) = \kappa(A)^2,$$

where κ is the condition number.

6.1 Preferred Numerical Methods

Method	Main idea
Normal equations	Direct theoretical formula; may amplify conditioning problems
QR factorization	Uses an orthonormal basis; stable for standard full-rank problems
SVD	Most informative for rank deficiency and ill-conditioning; usually more expensive

NumPy's `np.linalg.lstsq` uses a robust factorization and also reports the rank and singular values.

6.2 Rank Deficiency

If the columns of A are dependent, several parameter vectors may produce the same fitted values. A least-squares solver can return a minimum-norm solution, but coefficient interpretation becomes difficult.

6.3 Ridge Regularization Preview

Ridge regression minimizes

$$\|\mathbf{b} - A\mathbf{x}\|_2^2 + \lambda\|\mathbf{x}\|_2^2, \qquad \lambda > 0.$$

Its normal equations are

$$(A^T A + \lambda I)\hat{\mathbf{x}} = A^T\mathbf{b}.$$

Regularization trades a small amount of bias for reduced coefficient magnitude and improved stability.

Machine-Learning Connection

Modern ML practice cares about both approximation and generalization. Least squares explains the data fit; regularization and validation help control how the fitted model behaves on unseen data.

7 Python and NumPy Practice

7.1 Fitting a Straight Line

```python
import numpy as np

x = np.array([1.0, 2.0, 3.0])
y = np.array([2.0, 2.5, 4.0])

A = np.column_stack((np.ones_like(x), x))
beta, residual_sum, rank, singular_values = \
    np.linalg.lstsq(A, y, rcond=None)

y_hat = A @ beta
residuals = y - y_hat

print("Intercept and slope:", beta)
print("Predictions:", y_hat)
print("Residuals:", residuals)
print("Rank:", rank)
```

7.2 Calculating Regression Metrics

```python
sse = np.sum(residuals ** 2)
mse = np.mean(residuals ** 2)
rmse = np.sqrt(mse)
sst = np.sum((y - np.mean(y)) ** 2)
r_squared = 1 - sse / sst

print("SSE:", sse)
print("MSE:", mse)
print("RMSE:", rmse)
print("R-squared:", r_squared)
```

7.3 Visualizing the Fitted Line

```python
import matplotlib.pyplot as plt

plt.scatter(x, y, label="Observed data")
plt.plot(x, y_hat, color="purple", label="Least-squares line")
plt.xlabel("Input x")
plt.ylabel("Target y")
plt.legend()
plt.show()
```

> The `residual_sum` returned by `lstsq` may be empty for an exactly determined or rank-deficient system. Calculate residuals directly when you need consistent diagnostics.

8 ML Case Study: Sensor Calibration

A temperature sensor may have systematic offset and scaling error. Using reference measurements, fit

$$\text{true temperature} = \beta_0 + \beta_1(\text{sensor reading}).$$

```python
sensor = np.array([20.8, 25.4, 30.5, 35.3, 40.7])
true_temp = np.array([20.0, 25.0, 30.0, 35.0, 40.0])

A = np.column_stack((np.ones_like(sensor), sensor))
beta, _, rank, _ = np.linalg.lstsq(
    A, true_temp, rcond=None
)

calibrated = A @ beta
errors = true_temp - calibrated

print("Calibration coefficients:", beta)
print("Calibrated values:", calibrated)
print("RMSE:", np.sqrt(np.mean(errors ** 2)))
print("Rank:", rank)
```

8.1 Using the Calibration Model

```python
new_sensor_reading = 33.2
estimated_true_temp = (
    beta[0] + beta[1] * new_sensor_reading
)

print("Estimated true temperature:",
      estimated_true_temp)
```

8.2 Interpreting the Model

- β_0 estimates systematic offset.
- β_1 corrects the sensor's scale.
- Residuals measure calibration error at the reference points.
- Predictions outside the calibration range are extrapolations and require caution.

Machine-Learning Connection

This is supervised learning: reference temperatures are targets, sensor readings are inputs, and the fitted parameters define the prediction rule. The same design-matrix method extends to many features.

Least squares is sensitive to large outliers because errors are squared. Data quality checks and robust regression may be necessary when sensors occasionally produce abnormal readings.

9 Review and Independent Practice

Practice Tasks

1. Explain why an overdetermined system may not have an exact solution.
2. Derive the normal equations from the condition that the residual is orthogonal to Col(A).
3. Fit a line to $(0, 1)$, $(1, 2)$, $(2, 2)$, and $(3, 4)$ using NumPy. Report the intercept, slope, and RMSE.
4. Verify numerically that $A^T(\mathbf{y} - A\widehat{\boldsymbol{\beta}}) = \mathbf{0}$ for your fitted model.
5. Build a quadratic design matrix with columns 1, x, and x^2, then fit a quadratic curve.
6. Create a rank-deficient design matrix by duplicating a feature. Inspect the rank returned by `np.linalg.lstsq`.
7. Explain why validation error is more informative than training error when comparing model complexity.

9.1 Key Takeaways

- Least squares finds the parameter vector with the smallest residual norm.
- Fitted values are projections onto the column space of the design matrix.
- The normal equations express the orthogonality of the optimal residual.
- Linear and polynomial regression are matrix least-squares problems.
- Residual metrics measure fit, but validation is required for generalization.
- QR- and SVD-based solvers are usually preferable to explicitly forming an inverse.
- Rank deficiency creates non-unique coefficients and complicates interpretation.

9.2 Key Terms

Least squares	Residual	Design matrix
Normal equations	Regression coefficient	Projection
SSE	RMSE	R^2
Rank deficiency	Condition number	Regularization

Next lecture: Eigenvalues, eigenvectors, diagonalization, and stable transformation directions.