

STAC67: Regression Analysis

Lecture 9

Thibault Randrianarisoa

October 7, 2026

University of Toronto Scarborough

Announcement: the midterm date is fixed

Saturday 24 October, 1:00 to 3:30 pm, IA 1150

- A handwritten exam, on paper. Details on what it covers will follow.
- You may bring **one aid sheet**, written on both sides.
- It must be **handwritten**, with a pen on paper: not typed, not printed, not written on a tablet and printed.
- The final exam will allow the same format for the aid sheet. To reuse this one for the final, leave room on it for the material that comes after the midterm.

Announcement: Assignment 2 is out

Friday 23

Due ~~Wednesday~~ 21 October, 23:59, on Crowdmark. No late submissions.

- Six questions, each uploaded to **its own slot** on Crowdmark.
- Questions 1 and 6 in R Markdown: submit the knitted output, with code and output visible.
- Questions 2 to 5 typed or handwritten, as legible scans.
- Individual work, and no generative AI, as for Assignment 1.

 Uploading is not submitting: press **Submit** before the deadline.

Review of Lecture 8

Checking the four assumptions with the **residuals**:

- against the fitted values: a curve questions (L), a funnel questions (E), isolated points are outliers,
- a normal Q-Q plot for (N); a plot against time, or the order of collection, for (I).

Review of Lecture 8

Checking the four assumptions with the **residuals**:

- against the fitted values: a curve questions (L), a funnel questions (E), isolated points are outliers,
- a normal Q-Q plot for (N); a plot against time, or the order of collection, for (I).

Tests complete the plots, they do not replace them: **Shapiro-Wilk** for normality, **Brown-Forsythe** for constant variance.

Review of Lecture 8

Checking the four assumptions with the **residuals**:

- against the fitted values: a curve questions (L), a funnel questions (E), isolated points are outliers,
- a normal Q-Q plot for (N); a plot against time, or the order of collection, for (I).

Tests complete the plots, they do not replace them: **Shapiro-Wilk** for normality, **Brown-Forsythe** for constant variance.

When an assumption fails: transform Y (the **Box-Cox** family) or even X , and check the shape of the mean with a **lowess** curve.

Review of Lecture 8

Checking the four assumptions with the **residuals**:

- against the fitted values: a curve questions (L), a funnel questions (E), isolated points are outliers,
- a normal Q-Q plot for (N); a plot against time, or the order of collection, for (I).

Tests complete the plots, they do not replace them: **Shapiro-Wilk** for normality, **Brown-Forsythe** for constant variance.

When an assumption fails: transform Y (the **Box-Cox** family) or even X , and check the shape of the mean with a **lowess** curve.

Normality matters least for intervals on β_1 when n is large, and most for prediction intervals.

Matrices: what we need

Products and transposes

AB is defined when the number of **columns** of A equals the number of **rows** of B :

$$\underset{r \times c}{A} \underset{c \times s}{B} = \underset{r \times s}{C}, \quad \text{and in general } AB \neq BA.$$

Products and transposes

AB is defined when the number of **columns** of A equals the number of **rows** of B :

$$\underset{r \times c}{A} \underset{c \times s}{B} = \underset{r \times s}{C}, \quad \text{and in general } AB \neq BA.$$

Transposing swaps rows and columns, and reverses products: $(AB)' = B'A'$.

Products and transposes

AB is defined when the number of **columns** of A equals the number of **rows** of B :

$$\underset{r \times c}{A} \underset{c \times s}{B} = \underset{r \times s}{C}, \quad \text{and in general } AB \neq BA.$$

Transposing swaps rows and columns, and reverses products: $(AB)' = B'A'$.

In simple linear regression, with Y of size $n \times 1$ and the design matrix X of size $n \times 2$:

design matrix

$$X = \begin{bmatrix} 1 & X_1 \\ \vdots & \vdots \\ 1 & X_n \end{bmatrix}, \quad \text{m rows}$$
$$X'X = \begin{bmatrix} n & \sum X_i \\ \sum X_i & \sum X_i^2 \end{bmatrix}, \quad Y = \begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix}$$
$$X'Y = \begin{bmatrix} \sum Y_i \\ \sum X_i Y_i \end{bmatrix}, \quad Y'Y = \sum Y_i^2.$$

Products and transposes

AB is defined when the number of **columns** of A equals the number of **rows** of B :

$$\underset{r \times c}{A} \underset{c \times s}{B} = \underset{r \times s}{C}, \quad \text{and in general } AB \neq BA.$$

Transposing swaps rows and columns, and reverses products: $(AB)' = B'A'$.

In simple linear regression, with Y of size $n \times 1$ and the design matrix X of size $n \times 2$:

$$X = \begin{bmatrix} 1 & X_1 \\ \vdots & \vdots \\ 1 & X_n \end{bmatrix}, \quad X'X = \begin{bmatrix} n & \sum X_i \\ \sum X_i & \sum X_i^2 \end{bmatrix},$$

$$X'Y = \begin{bmatrix} \sum Y_i \\ \sum X_i Y_i \end{bmatrix}, \quad Y'Y = \sum Y_i^2.$$

$(X'X)' = (X')'(X')' = X'X$

$X'X$ is symmetric: it equals its own transpose.

Inverses

The **identity** I has 1 on the diagonal and 0 elsewhere:

$$I = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$= \# \text{ cols} = \# \text{ rows}$ $IA = AI = A.$

A square matrix of **full rank** (linearly independent columns) has an **inverse**:

$$A^{-1}A = AA^{-1} = I.$$

Inverses

The **identity** I has 1 on the diagonal and 0 elsewhere:

$$IA = AI = A.$$

A square matrix of **full rank** (linearly independent columns) has an **inverse**:

$$A^{-1}A = AA^{-1} = I.$$

For a 2×2 matrix:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{\underbrace{ad - bc}_{\substack{\text{determinant} \\ \text{of the matrix}}}} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}, \quad \text{so} \quad (X'X)^{-1} = \begin{bmatrix} \frac{1}{n} + \frac{\bar{X}^2}{S_{xx}} & -\frac{\bar{X}}{S_{xx}} \\ -\frac{\bar{X}}{S_{xx}} & \frac{1}{S_{xx}} \end{bmatrix}.$$

Inverses

The **identity** I has 1 on the diagonal and 0 elsewhere:

$$IA = AI = A.$$

A square matrix of **full rank** (linearly independent columns) has an **inverse**:

$$A^{-1}A = AA^{-1} = I.$$

For a 2×2 matrix:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}, \quad \text{so} \quad (\mathbf{X}'\mathbf{X})^{-1} = \begin{bmatrix} \frac{1}{n} + \frac{\bar{X}^2}{S_{xx}} & -\frac{\bar{X}}{S_{xx}} \\ -\frac{\bar{X}}{S_{xx}} & \frac{1}{S_{xx}} \end{bmatrix}.$$

It exists as long as $S_{xx} > 0$, that is, as long as the X_i are not all equal.

Inverses

The **identity** I has 1 on the diagonal and 0 elsewhere:

$$IA = AI = A.$$

A square matrix of **full rank** (linearly independent columns) has an **inverse**:

$$A^{-1}A = AA^{-1} = I.$$

For a 2×2 matrix:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}, \quad \text{so} \quad (\mathbf{X}'\mathbf{X})^{-1} = \begin{bmatrix} \frac{1}{n} + \frac{\bar{X}^2}{S_{xx}} & -\frac{\bar{X}}{S_{xx}} \\ -\frac{\bar{X}}{S_{xx}} & \frac{1}{S_{xx}} \end{bmatrix}.$$

It exists as long as $S_{xx} > 0$, that is, as long as the X_i are not all equal.

Rules: $(AB)^{-1} = B^{-1}A^{-1}$, $(A')^{-1} = (A^{-1})'$, the inverse of a symmetric matrix is symmetric.

Random vectors

For a random vector $\mathbf{Y} = (Y_1, \dots, Y_n)'$, the **mean** is the vector $E(\mathbf{Y})$ of the means $E(Y_i)$, and the **variance-covariance matrix** is

$$\sigma^2\{\mathbf{Y}\} = E[(\mathbf{Y} - E\mathbf{Y})(\mathbf{Y} - E\mathbf{Y})'],$$

with the variances $Var(Y_i)$ on the diagonal and the covariances $Cov(Y_i, Y_j)$ off it.

$$\mathbf{Y} = \begin{pmatrix} Y_1 \\ \vdots \\ Y_n \end{pmatrix} \quad \sigma^2\{\mathbf{Y}\} = \begin{pmatrix} Var(Y_1) & Cov(Y_1, Y_2) & \dots & Cov(Y_1, Y_n) \\ Cov(Y_2, Y_1) & Var(Y_2) & \dots & Cov(Y_2, Y_n) \\ \vdots & \vdots & \ddots & \vdots \\ Cov(Y_n, Y_1) & Cov(Y_n, Y_2) & \dots & Var(Y_n) \end{pmatrix}$$

Random vectors

For a random vector $\mathbf{Y} = (Y_1, \dots, Y_n)'$, the **mean** is the vector $E(\mathbf{Y})$ of the means $E(Y_i)$, and the **variance-covariance matrix** is

$$\sigma^2\{\mathbf{Y}\} = E[(\mathbf{Y} - E\mathbf{Y})(\mathbf{Y} - E\mathbf{Y})'],$$

with the variances $Var(Y_i)$ on the diagonal and the covariances $Cov(Y_i, Y_j)$ off it.

For two random vectors $\mathbf{U}$ and $\mathbf{V}$, the **covariance matrix between them** is

$$\sigma\{\mathbf{U}, \mathbf{V}\} = E[(\mathbf{U} - E\mathbf{U})(\mathbf{V} - E\mathbf{V})'], \quad \subset \sigma(\mathbf{V}, \mathbf{V})^T$$

whose entry (i, j) is $Cov(U_i, V_j)$. In particular, $\sigma\{\mathbf{Y}, \mathbf{Y}\} = \sigma^2\{\mathbf{Y}\}$.

Random vectors

For a random vector $\mathbf{Y} = (Y_1, \dots, Y_n)'$, the **mean** is the vector $E(\mathbf{Y})$ of the means $E(Y_i)$, and the **variance-covariance matrix** is

$$\sigma^2\{\mathbf{Y}\} = E[(\mathbf{Y} - E\mathbf{Y})(\mathbf{Y} - E\mathbf{Y})'],$$

with the variances $Var(Y_i)$ on the diagonal and the covariances $Cov(Y_i, Y_j)$ off it.

For two random vectors $\mathbf{U}$ and $\mathbf{V}$, the **covariance matrix between them** is

$$\sigma\{\mathbf{U}, \mathbf{V}\} = E[(\mathbf{U} - E\mathbf{U})(\mathbf{V} - E\mathbf{V})'],$$

whose entry (i, j) is $Cov(U_i, V_j)$. In particular, $\sigma\{\mathbf{Y}, \mathbf{Y}\} = \sigma^2\{\mathbf{Y}\}$.

With independent errors of variance σ^2 : $\sigma^2\{\boldsymbol{\epsilon}\} = \sigma^2 \mathbf{I}$.

$$\boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{pmatrix}$$

Linear functions of a random vector

For a matrix A of constants:

$$E(AY) = A E(Y), \quad \sigma^2\{AY\} = A \sigma^2\{Y\} A',$$

and, for a second matrix B of constants,

$$\sigma\{AY, BY\} = A \sigma^2\{Y\} B'.$$

Linear functions of a random vector

For a matrix A of constants:

$$E(AY) = A E(Y), \quad \sigma^2\{AY\} = A \sigma^2\{Y\} A',$$

and, for a second matrix B of constants,

$$\sigma\{AY, BY\} = A \sigma^2\{Y\} B'.$$

Why? Because $AY - E(AY) = A(Y - EY)$, so

$$\sigma\{AY, BY\} = E[A(Y - EY)(Y - EY)'B'] = A \sigma^2\{Y\} B'.$$

Linear functions of a random vector

For a matrix A of constants:

$$E(AY) = A E(Y), \quad \sigma^2\{AY\} = A \sigma^2\{Y\} A',$$

and, for a second matrix B of constants,

$$\sigma\{AY, BY\} = A \sigma^2\{Y\} B'.$$

$$\begin{aligned}\sigma(Y) &= \sigma(\varepsilon) \\ &= \sigma^2 I\end{aligned}$$

Why? Because $AY - E(AY) = A(Y - EY)$, so

$$\sigma\{AY, BY\} = E[A(Y - EY)(Y - EY)'B'] = A \sigma^2\{Y\} B'.$$

$$\begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}$$

Example. $Y_1, \dots, Y_n$ uncorrelated, mean μ , variance σ^2 . Then $\bar{Y} = \frac{1}{n} \mathbf{1}' Y$, and

$$\begin{aligned}\text{Var}(Y_i) &= \sigma^2 \\ \text{Corr}(Y_i, Y_j) &= 0 \\ &\text{if } i \neq j\end{aligned}$$

$$\text{Var}(\bar{Y}) = \frac{1}{n} \mathbf{1}' \sigma^2 I \frac{1}{n} \mathbf{1} = \frac{\sigma^2}{n^2} \mathbf{1}' \mathbf{1} = \frac{\sigma^2}{n}.$$

$$\Rightarrow n$$

Linear functions of a random vector

For a matrix A of constants:

$$E(AY) = A E(Y), \quad \sigma^2\{AY\} = A \sigma^2\{Y\} A',$$

and, for a second matrix B of constants,

$$\sigma\{AY, BY\} = A \sigma^2\{Y\} B'.$$

Why? Because $AY - E(AY) = A(Y - EY)$, so

$$\sigma\{AY, BY\} = E[A(Y - EY)(Y - EY)'B'] = A \sigma^2\{Y\} B'.$$

Example. $Y_1, \dots, Y_n$ uncorrelated, mean μ , variance σ^2 . Then $\bar{Y} = \frac{1}{n} \mathbf{1}'Y$, and

$$\text{Var}(\bar{Y}) = \frac{1}{n} \mathbf{1}' \sigma^2 I \frac{1}{n} \mathbf{1} = \frac{\sigma^2}{n^2} \mathbf{1}' \mathbf{1} = \frac{\sigma^2}{n}.$$

These rules do almost all the work in the rest of the course.

Simple linear regression in matrix form

The model and least squares

$$\beta = \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix}$$

$$x_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

$$Y = X\beta + \epsilon, \quad E(\epsilon) = \mathbf{0}, \quad \sigma^2\{\epsilon\} = \sigma^2 I,$$

and with normal errors, $\epsilon \sim N(\mathbf{0}, \sigma^2 I)$.

The model and least squares

$$Y = X\beta + \epsilon, \quad E(\epsilon) = 0, \quad \sigma^2\{\epsilon\} = \sigma^2 I,$$

and with normal errors, $\epsilon \sim N(0, \sigma^2 I)$.

Least squares minimizes $Q(\beta) = (Y - X\beta)'(Y - X\beta)$. The normal equations of Lecture 3 read

$$X'X \hat{\beta} = X'Y, \quad \text{so} \quad \hat{\beta} = (X'X)^{-1} X'Y.$$

$$\begin{pmatrix} n & \sum x_i \\ \sum x_i & \sum x_i^2 \end{pmatrix} \begin{pmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \end{pmatrix} = \begin{pmatrix} n\hat{\beta}_0 + \hat{\beta}_1 \sum x_i \\ \hat{\beta}_0 \sum x_i + \hat{\beta}_1 \sum x_i^2 \end{pmatrix}$$
$$\begin{pmatrix} 1 & \dots & 1 \\ x_1 & \dots & x_n \end{pmatrix} \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} \sum y_i \\ \sum x_i y_i \end{pmatrix}$$

The model and least squares

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \quad E(\boldsymbol{\epsilon}) = \mathbf{0}, \quad \sigma^2\{\boldsymbol{\epsilon}\} = \sigma^2\mathbf{I},$$

and with normal errors, $\boldsymbol{\epsilon} \sim N(\mathbf{0}, \sigma^2\mathbf{I})$.

Least squares minimizes $Q(\boldsymbol{\beta}) = (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})'(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})$. The normal equations of Lecture 3 read

$$\mathbf{X}'\mathbf{X} \hat{\boldsymbol{\beta}} = \mathbf{X}'\mathbf{Y}, \quad \text{so} \quad \hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1} \mathbf{X}'\mathbf{Y}.$$

With the inverse of the previous section, its two rows give back $\hat{\beta}_1 = S_{xY}/S_{xx}$ and $\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1\bar{X}$.

The model and least squares

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \quad E(\boldsymbol{\epsilon}) = \mathbf{0}, \quad \sigma^2\{\boldsymbol{\epsilon}\} = \sigma^2\mathbf{I},$$

and with normal errors, $\boldsymbol{\epsilon} \sim N(\mathbf{0}, \sigma^2\mathbf{I})$.

Least squares minimizes $Q(\boldsymbol{\beta}) = (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})'(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})$. The normal equations of Lecture 3 read

$$\mathbf{X}'\mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}'\mathbf{Y}, \quad \text{so} \quad \hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}.$$

With the inverse of the previous section, its two rows give back $\hat{\beta}_1 = S_{xY}/S_{xx}$ and $\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1\bar{X}$.

The formula $\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$ stays the same with several predictors: only $\mathbf{X}$ gets more columns.

Fitted values and residuals

The fitted values are a linear function of Y :

$$\hat{Y} = X\hat{\beta} = X(X'X)^{-1}X'Y = HY, \quad H = X(X'X)^{-1}X'.$$

H , of size $n \times n$, is the **hat matrix**: it puts the hat on Y .

Fitted values and residuals

The fitted values are a linear function of Y :

$$\hat{Y} = X\hat{\beta} = X(X'X)^{-1}X'Y = HY, \quad H = X(X'X)^{-1}X'.$$

H , of size $n \times n$, is the **hat matrix**: it puts the hat on Y .

So are the residuals:

$$e = Y - \hat{Y} = (I - H)Y, \quad SSE = e'e, \quad MSE = \frac{e'e}{n - 2}.$$

Fitted values and residuals

The fitted values are a linear function of Y :

$$\hat{Y} = X\hat{\beta} = X(X'X)^{-1}X'Y = HY, \quad H = X(X'X)^{-1}X'.$$

H , of size $n \times n$, is the **hat matrix**: it puts the hat on Y .

So are the residuals:

$$e = Y - \hat{Y} = (I - H)Y, \quad SSE = e'e, \quad MSE = \frac{e'e}{n - 2}.$$

H is symmetric and **idempotent**: $HH = H$. Applying it twice changes nothing.

Fitted values and residuals

The fitted values are a linear function of Y :

$$\hat{Y} = X\hat{\beta} = X(X'X)^{-1}X'Y = HY, \quad H = X(X'X)^{-1}X'.$$

H , of size $n \times n$, is the **hat matrix**: it puts the hat on Y .

So are the residuals:

$$e = Y - \hat{Y} = (I - H)Y, \quad SSE = e'e, \quad MSE = \frac{e'e}{n - 2}.$$

H is symmetric and **idempotent**: $HH = H$. Applying it twice changes nothing.

You will prove this in Assignment 2, and see what it implies for the residuals.

The mean and variance of $\hat{\beta}$

$\hat{\beta} = \mathbf{A}\mathbf{Y}$ with the constant matrix $\mathbf{A} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$. Apply the rules:

The mean and variance of $\hat{\beta}$

$\hat{\beta} = \mathbf{A}\mathbf{Y}$ with the constant matrix $\mathbf{A} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$. Apply the rules:

$$E(\hat{\beta}) = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}' E(\mathbf{Y}) = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{X} \beta = \beta : \quad \text{unbiased.}$$

The mean and variance of $\hat{\beta}$

$\hat{\beta} = \mathbf{A}\mathbf{Y}$ with the constant matrix $\mathbf{A} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$. Apply the rules:

$$E(\hat{\beta}) = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}' E(\mathbf{Y}) = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{X} \beta = \beta : \quad \text{unbiased.}$$

$$\sigma^2\{\hat{\beta}\} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}' \sigma^2 \mathbf{I} \mathbf{X} (\mathbf{X}'\mathbf{X})^{-1}$$

The mean and variance of $\hat{\beta}$

$\hat{\beta} = \mathbf{A}\mathbf{Y}$ with the constant matrix $\mathbf{A} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$. Apply the rules:

$$E(\hat{\beta}) = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}' E(\mathbf{Y}) = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{X} \beta = \beta : \quad \text{unbiased.}$$

$$\begin{aligned}\sigma^2\{\hat{\beta}\} &= (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}' \sigma^2 \mathbf{I} \mathbf{X} (\mathbf{X}'\mathbf{X})^{-1} \\ &= \sigma^2 (\mathbf{X}'\mathbf{X})^{-1} \underbrace{\mathbf{X}'\mathbf{X} (\mathbf{X}'\mathbf{X})^{-1}}_{\mathbf{I}} = \sigma^2 (\mathbf{X}'\mathbf{X})^{-1}.\end{aligned}$$

The mean and variance of $\hat{\beta}$

$\hat{\beta} = \mathbf{A}\mathbf{Y}$ with the constant matrix $\mathbf{A} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$. Apply the rules:

$$E(\hat{\beta}) = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}' E(\mathbf{Y}) = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{X} \beta = \beta : \quad \text{unbiased.}$$

$$\begin{aligned}\sigma^2\{\hat{\beta}\} &= (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}' \sigma^2 \mathbf{I} \mathbf{X} (\mathbf{X}'\mathbf{X})^{-1} \\ &= \sigma^2 (\mathbf{X}'\mathbf{X})^{-1} \underbrace{\mathbf{X}'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}}_{\mathbf{I}} = \sigma^2 (\mathbf{X}'\mathbf{X})^{-1}.\end{aligned}$$

We used $((\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}')' = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}$, since $(\mathbf{X}'\mathbf{X})^{-1}$ is symmetric.

$$\sigma^2\{\hat{\beta}\} = \sigma^2(\mathbf{X}'\mathbf{X})^{-1} = \begin{bmatrix} \sigma^2\left(\frac{1}{n} + \frac{\bar{X}^2}{S_{xx}}\right) & -\frac{\bar{X}\sigma^2}{S_{xx}} \\ -\frac{\bar{X}\sigma^2}{S_{xx}} & \frac{\sigma^2}{S_{xx}} \end{bmatrix} \begin{matrix} \leftarrow \text{Var}(\hat{\beta}_0), \text{Cov}(\hat{\beta}_0, \hat{\beta}_1) \\ \leftarrow \text{Cov}(\hat{\beta}_0, \hat{\beta}_1), \text{Var}(\hat{\beta}_1) \end{matrix}$$

$$\sigma^2\{\hat{\beta}\} = \sigma^2(\mathbf{X}'\mathbf{X})^{-1} = \begin{bmatrix} \sigma^2\left(\frac{1}{n} + \frac{\bar{X}^2}{S_{xx}}\right) & -\frac{\bar{X}\sigma^2}{S_{xx}} \\ -\frac{\bar{X}\sigma^2}{S_{xx}} & \frac{\sigma^2}{S_{xx}} \end{bmatrix} \begin{matrix} \leftarrow \text{Var}(\hat{\beta}_0), \text{Cov}(\hat{\beta}_0, \hat{\beta}_1) \\ \leftarrow \text{Cov}(\hat{\beta}_0, \hat{\beta}_1), \text{Var}(\hat{\beta}_1) \end{matrix}$$

All three formulas of Lecture 4 are collected in this matrix.

Reading $\sigma^2(\mathbf{X}'\mathbf{X})^{-1}$

$$\sigma^2\{\hat{\beta}\} = \sigma^2(\mathbf{X}'\mathbf{X})^{-1} = \begin{bmatrix} \sigma^2\left(\frac{1}{n} + \frac{\bar{X}^2}{S_{xx}}\right) & -\frac{\bar{X}\sigma^2}{S_{xx}} \\ -\frac{\bar{X}\sigma^2}{S_{xx}} & \frac{\sigma^2}{S_{xx}} \end{bmatrix} \begin{matrix} \leftarrow \text{Var}(\hat{\beta}_0), \text{Cov}(\hat{\beta}_0, \hat{\beta}_1) \\ \leftarrow \text{Cov}(\hat{\beta}_0, \hat{\beta}_1), \text{Var}(\hat{\beta}_1) \end{matrix}$$

All three formulas of Lecture 4 are collected in this matrix.

Replacing σ^2 by MSE gives the **estimated** variance-covariance matrix

$$s^2\{\hat{\beta}\} = MSE (\mathbf{X}'\mathbf{X})^{-1},$$

and the standard errors $s\{\hat{\beta}_0\}$, $s\{\hat{\beta}_1\}$ are the square roots of its diagonal.

In R: $\hat{\beta}$ from the matrices

We consider the crime rate data of Lectures 4 to 6, with `fit <- lm(Y ~ X)`.

- `model.matrix(fit)` builds the design matrix X : a column of 1s, then the column of X_i ;
- `t(A)` is the transpose A' , `A %*% B` the matrix product AB , `solve(A)` the inverse A^{-1} .

In R: $\hat{\beta}$ from the matrices

We consider the crime rate data of Lectures 4 to 6, with `fit <- lm(Y ~ X)`.

- `model.matrix(fit)` builds the design matrix X : a column of 1s, then the column of X_i ;
- `t(A)` is the transpose A' , `A %*% B` the matrix product AB , `solve(A)` the inverse A^{-1} .

```
X <- model.matrix(fit)
XtX <- t(X) %*% X
beta_hat <- solve(XtX) %*% t(X) %*% Crime$Y
cbind(manual_fit = drop(beta_hat), lm = coef(fit))
```

```
##                manual_fit                lm
## (Intercept) 20517.5999 20517.5999
## X           -170.5752  -170.5752
```

In R: $\hat{\beta}$ from the matrices

We consider the crime rate data of Lectures 4 to 6, with `fit <- lm(Y ~ X)`.

- `model.matrix(fit)` builds the design matrix X : a column of 1s, then the column of X_i ;
- `t(A)` is the transpose A' , `A %*% B` the matrix product AB , `solve(A)` the inverse A^{-1} .

```
X <- model.matrix(fit)
XtX <- t(X) %*% X
beta_hat <- solve(XtX) %*% t(X) %*% Crime$Y
cbind(manual_fit = drop(beta_hat), lm = coef(fit))
```

```
##                manual_fit                lm
## (Intercept) 20517.5999 20517.5999
## X           -170.5752  -170.5752
```

We get the same estimates as `lm`, but computed with the formula

$$\hat{\beta} = (X'X)^{-1}X'Y.$$

In R: the estimated variance-covariance matrix

$MSE = SSE/(n - 2)$, then $s^2\{\hat{\beta}\} = MSE (\mathbf{X}'\mathbf{X})^{-1}$:

```
MSE <- sum(resid(fit)^2) / (nrow(X) - 2)
s2_beta <- MSE * solve(XtX)
s2_beta
```

```
##              (Intercept)              X
## (Intercept)    10742942 -135845.953
## X              -135846   1728.425
```

```
vcov(fit) # computed by lm
```

```
##              (Intercept)              X
## (Intercept)    10742942 -135845.953
## X              -135846   1728.425
```

In R: the estimated variance-covariance matrix

$MSE = SSE/(n - 2)$, then $s^2\{\hat{\beta}\} = MSE (\mathbf{X}'\mathbf{X})^{-1}$:

```
MSE <- sum(resid(fit)^2) / (nrow(X) - 2)
s2_beta <- MSE * solve(XtX)
s2_beta
```

```
##                (Intercept)                X
## (Intercept)    10742942 -135845.953
## X              -135846   1728.425
```

```
vcov(fit) # computed by lm
```

```
##                (Intercept)                X
## (Intercept)    10742942 -135845.953
## X              -135846   1728.425
```

The diagonal holds the estimated variances: $s\{\hat{\beta}_1\} = \sqrt{1728.4} = 41.57$, the standard error behind the crime rate t test of Lectures 5 and 6.

Several predictors

Models with several predictors

Most problems have **more than one** potential predictor. The aim is to find the effect of each one, **controlling for the others**.

Models with several predictors

Most problems have **more than one** potential predictor. The aim is to find the effect of each one, **controlling for the others**.

The **first-order model** with two numeric predictors:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \epsilon_i, \quad E(Y) = \beta_0 + \beta_1 X_1 + \beta_2 X_2.$$

The mean response is a **plane** above the (X_1, X_2) plane.

Models with several predictors

Most problems have **more than one** potential predictor. The aim is to find the effect of each one, **controlling for the others**.

The **first-order model** with two numeric predictors:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \epsilon_i, \quad E(Y) = \beta_0 + \beta_1 X_1 + \beta_2 X_2.$$

The mean response is a **plane** above the (X_1, X_2) plane.

The same framework will allow us to handle:

- polynomial terms X_i^p , for curved relations;
- products of predictors $X_{i1}X_{i2}$, when the effect of one depends on the level of another;
- discrete variables, for categorical predictors.

Example: acceleration of cars

We observe 390 cars in the 'autompg' dataset from the UCI Machine Learning Repository: acc, the time to accelerate from 0 to 60 mph (in seconds), against the weight wt (in lb) and the horsepower hp.

```
fit2 <- lm(acc ~ wt + hp, data = autompg)
round(coef(summary(fit2)), 5)
```

##	Estimate	Std. Error	t value	Pr(> t)
## (Intercept)	18.46959	0.32651	56.56593	0
## wt	0.00229	0.00021	11.10182	0
## hp	-0.09326	0.00455	-20.47774	0

Example: acceleration of cars

We observe 390 cars in the 'autmpg' dataset from the UCI Machine Learning Repository: acc, the time to accelerate from 0 to 60 mph (in seconds), against the weight wt (in lb) and the horsepower hp.

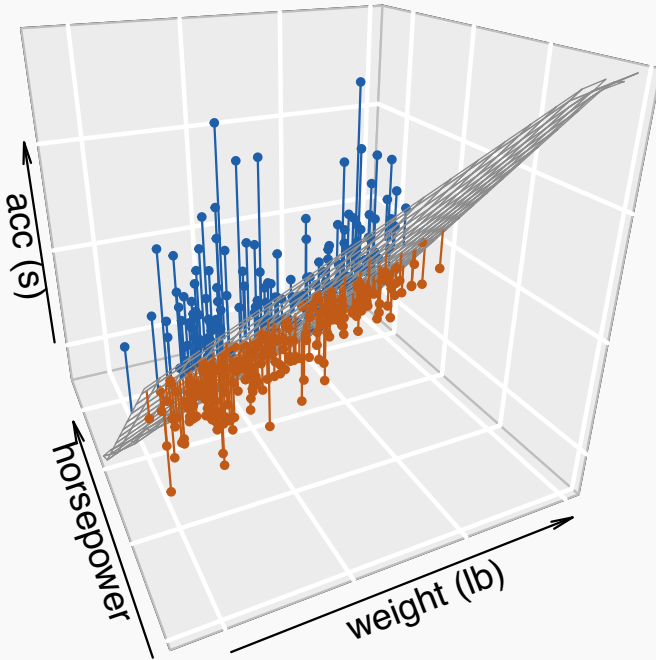
```
fit2 <- lm(acc ~ wt + hp, data = autmpg)
round(coef(summary(fit2)), 5)
```

##	Estimate	Std. Error	t value	Pr(> t)
## (Intercept)	18.46959	0.32651	56.56593	0
## wt	0.00229	0.00021	11.10182	0
## hp	-0.09326	0.00455	-20.47774	0

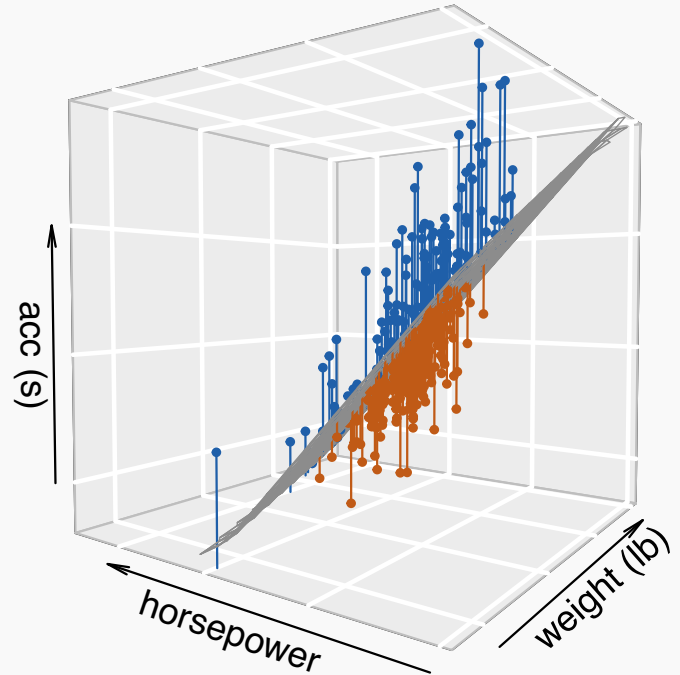
Reading $\hat{\beta}_1$: among cars with the same horsepower, each extra 1000 lb adds about 2.3 seconds to the mean time to 60 mph.

The fitted plane, from two angles

The plane seen from above



The plane seen edge on



Each vertical segment is a residual.

Why not two simple regressions?

Regressing acc on wt alone gives a slope of -1.36 seconds per 1000 lb: heavier cars seem faster.

Why not two simple regressions?

Regressing acc on wt alone gives a slope of -1.36 seconds per 1000 lb: **heavier cars seem faster.**

With hp in the model, the slope is +2.29 seconds per 1000 lb: **heavier cars are slower**, at a given horsepower.

Why not two simple regressions?

Regressing acc on wt alone gives a slope of -1.36 seconds per 1000 lb: **heavier cars seem faster**.

With hp in the model, the slope is +2.29 seconds per 1000 lb: **heavier cars are slower**, at a given horsepower.

The two predictors are strongly related,

$$r(\text{wt}, \text{hp}) = 0.86$$

Heavy cars tend to have big engines. Alone, the weight carries the effect of the engine with it.

Why not two simple regressions?

Regressing acc on wt alone gives a slope of -1.36 seconds per 1000 lb: **heavier cars seem faster.**

With hp in the model, the slope is +2.29 seconds per 1000 lb: **heavier cars are slower**, at a given horsepower.

The two predictors are strongly related,

$$r(\text{wt}, \text{hp}) = 0.86$$

Heavy cars tend to have big engines. Alone, the weight carries the effect of the engine with it.

A coefficient in a multiple regression is the effect of its predictor with the others held fixed. It can differ from the simple regression slope, even in sign.

Why not two simple regressions?

Regressing acc on wt alone gives a slope of -1.36 seconds per 1000 lb: **heavier cars seem faster**.

With hp in the model, the slope is +2.29 seconds per 1000 lb: **heavier cars are slower**, at a given horsepower.

The two predictors are strongly related,

$$r(\text{wt}, \text{hp}) = 0.86$$

Heavy cars tend to have big engines. Alone, the weight carries the effect of the engine with it.

A coefficient in a multiple regression is the effect of its predictor with the others held fixed. It can differ from the simple regression slope, even in sign.

Next: the model with $p - 1$ predictors in matrix form.