Chapter 3 Linear Regression

– Math 313 Statistics for Data Science

Guangliang Chen

Associate Professor

cheng@hope.edu

Hope College, Fall 2023

Presentation Overview

1. 3.1 Simple linear regression

2. 3.2 Multiple linear regression

3. 3.3 Other Considerations in the Regression Model

4. Optional: Linear regression via gradient descent

3.1 Simple linear regression

The simple linear regression problem

Assume a linear model

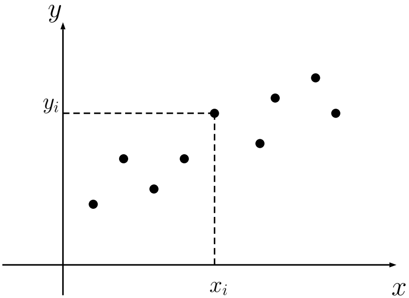
$$Y = \beta_0 + \beta_1 X + \epsilon$$

where $\epsilon \sim N(0, \sigma^2)$ and it is independent of X. The model parameters $\beta_0, \beta_1, \sigma^2$ are all unknown.

Given a set of n observations from the above model,

$$\{(x_i, y_i) \mid 1 \leq i \leq n\},$$

the goal of regression is to use the sample to estimate β_0, β_1 (and also σ^2 sometimes).



Regression models are only empirical models

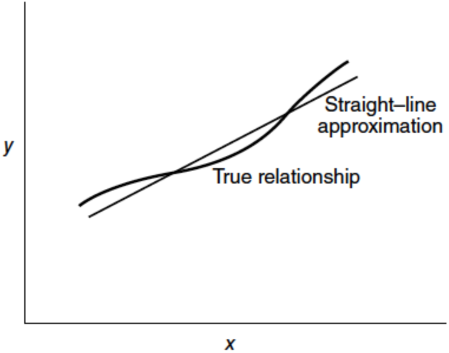


Figure 1.3 Linear regression approximation of a complex relationship.

Often the true model is nonlinear

$$Y = f(X) + \epsilon$$

When performing simple linear regression, we are attempting to approximate the nonlinear relationship by a linear one:

$$f(X) \approx \beta_0 + \beta_1 X$$

This only controls the reducible error (the irreducible error cannot be reduced).

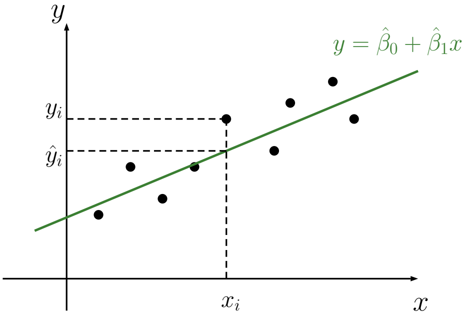
Least-squares (LS) estimation

Here we adopt the **least squares (LS)** criterion for estimating the coefficients β_0, β_1:

$$\min_{\hat\beta_0, \hat\beta_1} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

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

The minimizers $\hat\beta_0, \hat\beta_1$ are called **LS estimates** of β_0, β_1, and the fitted line is called the **LS regression line**.



y_i: observation, $\hat{y}_i$: fitted value

Notation

Denote by

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \quad \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

and

$$S_{xx} = \sum_{i=1}^{n}(x_i - \bar{x})^2$$

$$S_{xy} = \sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$

It can be shown that

$$S_{xx} = \sum_{i=1}^{n} x_i^2 - n\bar{x}^2,$$

$$S_{xy} = \sum_{i=1}^{n} x_i y_i - n\bar{x}\bar{y}$$

Result

Theorem

The LS estimators of the intercept and slope in the simple linear regression model are

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x},$$

$$\hat{\beta}_1 = \frac{S_{xy}}{S_{xx}}$$

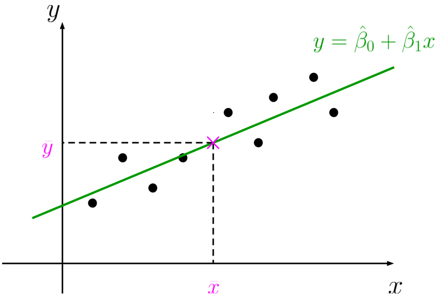
Some observations

- The LS regression line always passes through the centroid $(\bar{x}, \bar{y})$ of the data:

$$\bar{y} = \hat{\beta}_0 + \hat{\beta}_1 \bar{x}.$$

- An alternative form of the equation of the LS regression line is

$$y = \underbrace{(\bar{y} - \hat{\beta}_1 \bar{x})}_{\hat{\beta}_0} + \hat{\beta}_1 x$$
$$= \bar{y} + \hat{\beta}_1 (x - \bar{x})$$



- Let

$$e_i = y_i - \hat{y}_i = y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 x_i\right), \quad 1 \le i \le n$$

which are called the **residuals** of the model. It can be shown that they satisfy

$$\sum_{i=1}^{n} e_i = 0.$$

- It follows that

$$\sum_{i=1}^{n} y_i = \sum_{i=1}^{n} \hat{y}_i.$$

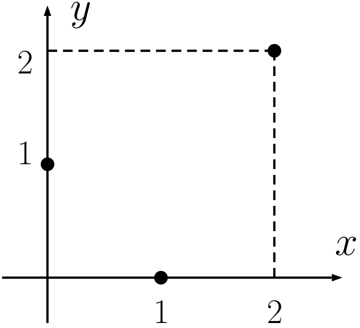
This implies that $\{\hat{y}_i\}$ and $\{y_i\}$ have the same mean, i.e., $\bar{y}$.

- The total least-squares fitting error is

$$\sum_{i=1}^{n} e_i^2.$$

Example (Toy data)

Given a data set of 3 points: $(0,1),(1,0),(2,2)$, find the LS regression line.



x_iy_i				$\sum x_iy_i =$
x_i^2				$\sum x_i^2 =$
x_i	0	1	2	$\sum x_i =$
y_i	1	0	2	$\sum y_i =$
$\hat{y}_i$				$\sum \hat{y}_i^2 =$
e_i				$\sum e_i^2 =$

Solution. First, $\bar{x} = 1 = \bar{y}$, and

$$S_{xx} = \sum x_i^2 - n\bar{x}^2 = 5 - 3 = 2, \quad S_{xy} = \sum x_i y_i - n\bar{x}\bar{y} = 4 - 3 = 1.$$

It follows that

$$\hat{\beta}_1 = \frac{S_{xy}}{S_{xx}} = \frac{1}{2}, \quad \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x} = \frac{1}{2}.$$

Thus, the regression line is given by

$$y = \hat{\beta}_0 + \hat{\beta}_1 x = \frac{1}{2} + \frac{1}{2}x.$$

The fitted values and their residuals are

$$\hat{y}_1 = \frac{1}{2}, \ \hat{y}_2 = 1, \ \hat{y}_3 = \frac{3}{2} \quad \text{and } e_1 = \frac{1}{2}, \ e_2 = -1, \ e_3 = \frac{1}{2}$$

Question: What is the PCA line (and is it the same with the LS regression line)?

Sums of squares

Define

- **Total Sum of Squares**:

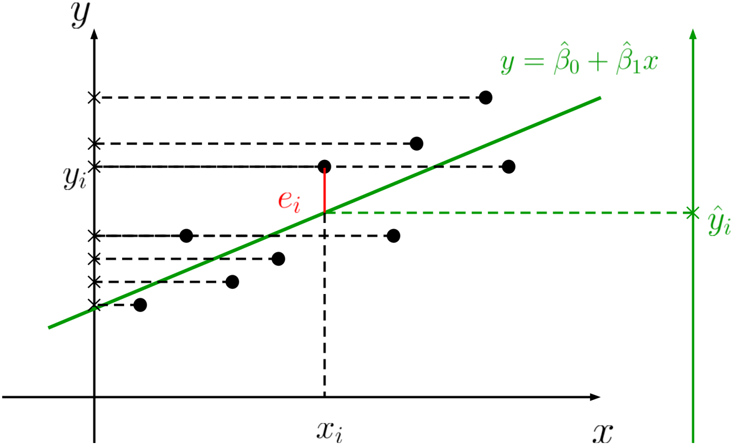
$$SS_T = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

- **Regression Sum of Squares**:

$$SS_R = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2$$

- **Residual Sum of Squares**:

$$SS_{Res} = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$



It can be shown that

$$SS_T = SS_R + SS_{Res}$$

which represents a decomposition of the total scatter of the response of the data into the scatter of the fitted values (SS_R) and the scatter of the residuals (SS_{Res}) .

Goodness of fit in simple linear regression

The total LS fitting error, SS_{Res}, is a measure of the goodness of fit of the regression line, but it depends on the scale (units) of the data.

A relative measure that is unit free is the following.

Definition (Coefficient of determination)

$$R^2 = \frac{SS_R}{SS_T} = 1 - \frac{SS_{Res}}{SS_T}$$

Remark. The quantity $0 \le R^2 \le 1$ indicates the relative amount of scatter of the response that is explained by the regression line.

The higher R^2, the better the fit.

Example (Cont'd)

Consider again the toy data set that consists of 3 points: $(0,1),(1,0),(2,2)$. We have fitted the LS regression line earlier. It follows that

$$SS_{Res} = \sum e_i^2 = \left(\frac{1}{2}\right)^2 + (-1)^2 + \left(\frac{1}{2}\right)^2 = \frac{3}{2}.$$

To compute the coefficient of determination, we also need to compute $SS_T = \sum y_i^2 - n\bar{y}^2 = 2$. Therefore,

$$R^2 = 1 - \frac{SS_{Res}}{SS_T} = 1 - \frac{1.5}{2} = 0.25$$

This is a poor fit.

Follow-up question: What is SS_R?

Model adequacy checking

The sample regression model is

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i, \quad 1 \le i \le n$$

where $\epsilon_1, \ldots, \epsilon_n$ are identically and independently distributed according to $N(0, \sigma^2)$.

The model assumptions, linearity, normality, and constant variance, all need to be verified in order for simple linear regression to be effective.

Next, we explain for a fitted model

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i, \quad 1 \le i \le n$$

how to use its “residuals” to check the adequacy of the model.

Residuals

The **residuals** of the fitted model are defined as

$$e_i = y_i - \hat{y}_i, \quad i = 1, \ldots, n$$

They are viewed as observations of the model errors ϵ_i, and thus can be used to check the model assumptions.

What we know about the residuals:

- The residuals always sum to zero: $\sum e_i = 0$
- They are also uncorrelated with the fitted values: $\sum \hat{y}_i e_i = 0$.
- They can be used to form an unbiased estimator for σ^2, called *residual mean square*:

$$MS_{Res} = \frac{SS_{Res}}{n-2} = \frac{1}{n-2} \sum_{i=1}^{n} e_i^2.$$

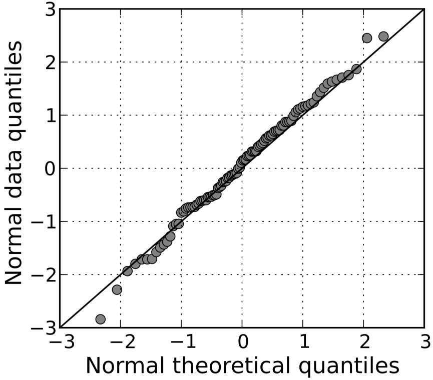
Residual plots

Graphical analysis is much more effective in trying to detect patterns in the residuals than looking at the raw numbers.

There are different types of plots that can be employed to check the different model assumptions.

- **Normal quantile plots (qq-plots)** $\longleftarrow$ checking normality
- **Residuals against fitted values** $\longleftarrow$ checking constant variance, or nonlinearity

Normal quantile plot for checking normality



What to look for: A generally linear pattern (small violations okay).

Significant departure from this pattern is strong evidence against the normality assumption.

Residuals against fitted values

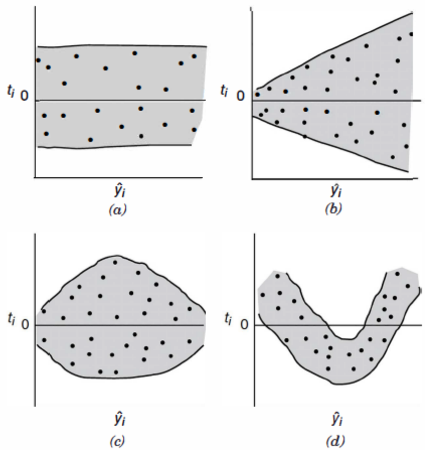
This plot may be used to check the constant-variance assumption of the model error (and also nonlinearity):

(a) Confetti in a box ✓

(b) Funnel X

(c) Double bow X

(d) Curvature

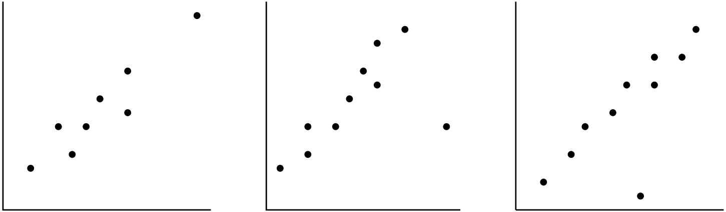


To fix these issues, one can transform the response and/or add a nonlinear form of the predictor (e.g., $x^2, \sqrt{x}, \log x$).

Outliers and high-leverage plots

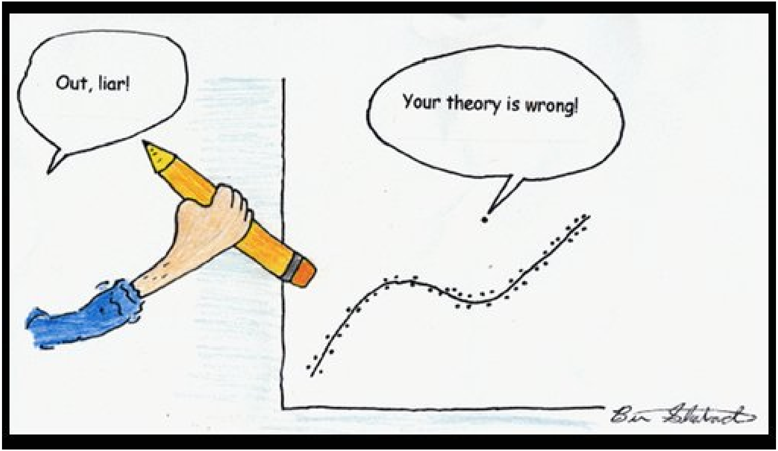
The residual plots can also reveal outliers and high-leverage plots.

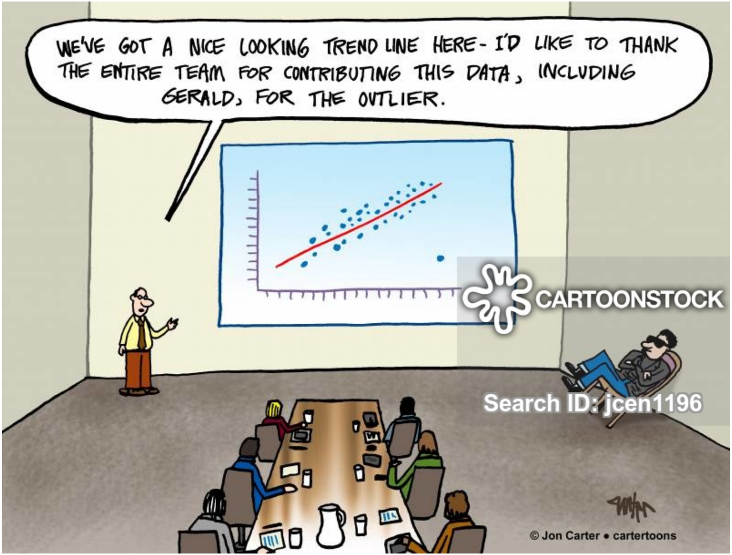
- An outlier is a point that is outlying in x-space, y-space, or both.
- A leverage point is an observation that has a very different predictor value from the bulk of the observations.



An outlier may or may not have high leverage, depending on its predictor value: $\text{leverageScore}(x_i) = \frac{1}{n} + \frac{(x_i - \bar{x})^2}{S_{xx}}$.

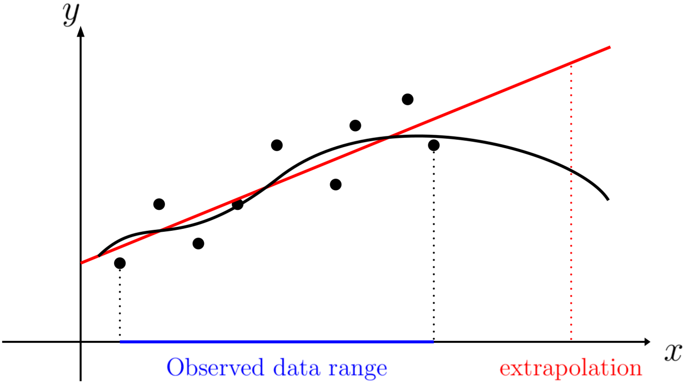
Do not remove outliers unless they are truly due to errors!

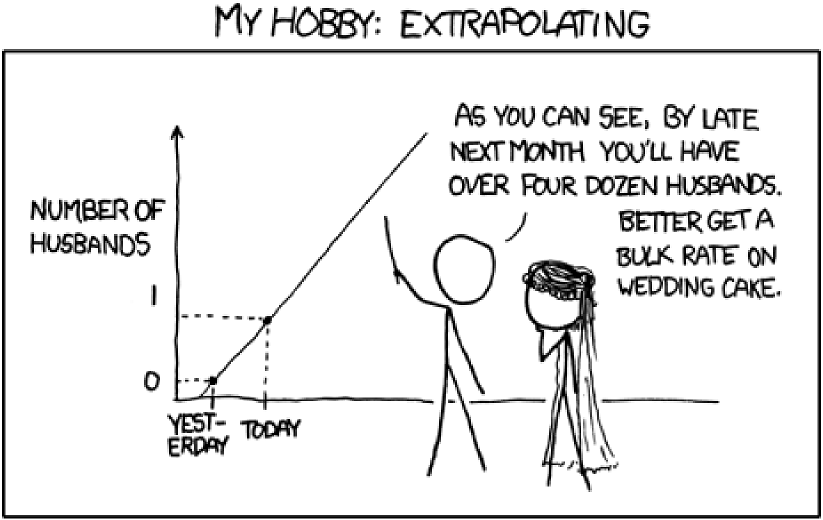

Out, liar!
Your theory is wrong!


WE'VE GOT A NICE LOOKING TREND LINE HERE - I'D LIKE TO THANK THE ENTIRE TEAM FOR CONTRIBUTING THIS DATA, INCLUDING GERALD, FOR THE OUTLIER.
CARTOONSTOCK
Search ID: jcen1196
© Jon Carter • cartertoons

Other considerations in the use of simple linear regression

- Do not extrapolate!

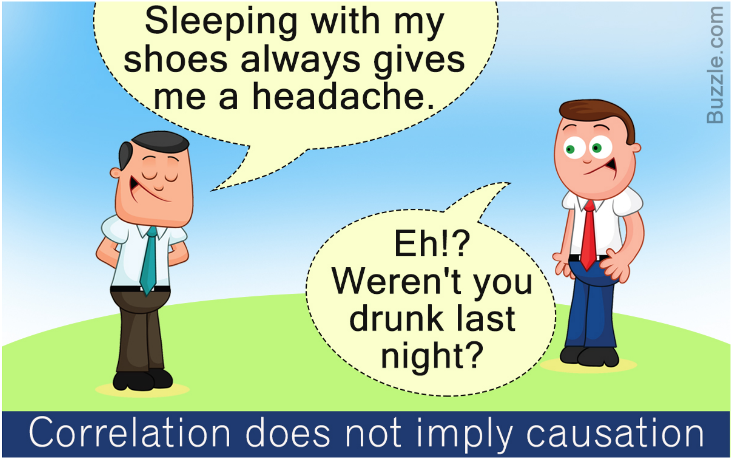



MY HOBBY: EXTRAPOLATING
NUMBER OF HUSBANDS
1
0
YEST-ERDAY
TODAY
AS YOU CAN SEE, BY LATE NEXT MONTH YOU'LL HAVE OVER FOUR DOZEN HUSBANDS.
BETTER GET A BULK RATE ON WEDDING CAKE.

WHY IS THAT WOMAN SCOWLING AT ME? DO I KNOW HER?

If she loves you more each and every day,
by linear regression she hated you before you met.

- Correlation does not imply causation


Sleeping with my shoes always gives me a headache.
Eh!? Weren't you drunk last night?
Buzzle.com
Correlation does not imply causation

3.2 Multiple linear regression

The multiple linear regression problem

Consider a linear model with multiple predictors

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k + \epsilon$$

where

- Y: response,
- $X_1, \ldots, X_k$: predictors
- $\epsilon \sim N(0, \sigma^2)$: noise ($\sigma^2$ often unknown)
- $\beta_0, \beta_1, \ldots, \beta_k$: coefficients (unknown)

Given n observations of the response and predictors,

$$(x_{i1}, x_{i2}, \ldots, x_{ik}, y_i), \quad 1 \leq i \leq n$$

the goal is to estimate the model parameters $\beta_0, \beta_1, \ldots, \beta_k$ (and σ^2).

Least squares (LS) estimation

The LS criterion can still be used to fit a multiple regression model

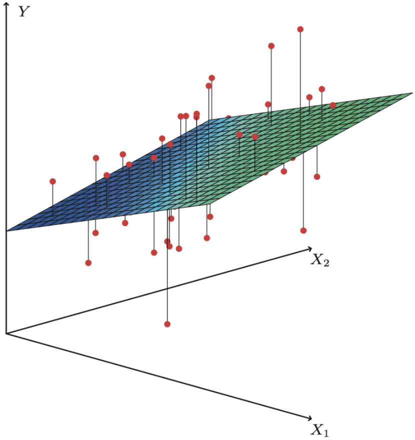
$$y = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_k x_k$$

to the data by solving

$$\min_{\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_k} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

where for each $1 \leq i \leq n$,

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \cdots + \hat{\beta}_k x_{ik}$$



Let $p = k + 1$ (#regression coefficients including the intercept).

Denote

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \in \mathbb{R}^n, \quad \mathbf{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1k} \\ 1 & x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nk} \end{bmatrix} \in \mathbb{R}^{n\times p}, \quad \hat{\boldsymbol{\beta}} = \begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \\ \vdots \\ \hat{\beta}_k \end{bmatrix} \in \mathbb{R}^p$$

which are called the response vector, the design matrix, and the vector of regression coefficients, respectively.

The least squares formulation of multiple linear regression can be rewritten as

$$\min_{\hat{\boldsymbol{\beta}}} \|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}\|^2$$

Theorem

The LS estimator of β satisfies the following linear system (called normal equation)

$$(\mathbf{X}^T\mathbf{X})\,\hat{\beta} = \mathbf{X}^T\mathbf{y}$$

Remark. If all the columns of **X** are linearly independent, then the above linear system has a unique solution:

$$\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

The LS fitted values are

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\beta} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

Example

Consider the same toy data set of 3 points: $(0,1),(1,0),(2,2)$. Use the general procedure for multiple linear regression to fit a regression line. Find also the fitted values and residuals.

Solution. The design matrix and response vector are

$$\mathbf{X} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix}$$

We form the following the normal equation

$$\left(\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix} \right) \hat{\beta} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix}$$

which can be simplified to

$$\begin{bmatrix} 3 & 3 \\ 3 & 5 \end{bmatrix} \hat{\beta} = \begin{bmatrix} 3 \\ 4 \end{bmatrix}$$

The system has a unique solution

$$\hat{\beta} = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}$$

It follows that the fitted values are

$$\hat{\mathbf{y}} = \underbrace{\begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{bmatrix}}_{\mathbf{X}} \underbrace{\begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}}_{\hat{\beta}} = \begin{bmatrix} 0.5 \\ 1 \\ 1.5 \end{bmatrix}$$

with residuals

$$\mathbf{e} = \mathbf{y} - \hat{\mathbf{y}} = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix} - \begin{bmatrix} 0.5 \\ 1 \\ 1.5 \end{bmatrix} = \begin{bmatrix} 0.5 \\ -1 \\ 0.5 \end{bmatrix}$$

Goodness of fit in multiple regression

R^2 measures the goodness of fit of a single model and is not a fair criterion for comparing models with different sizes k (e.g., nested models)

For example,

$$\begin{aligned} y &= \beta_0 + \beta_1 x_1, & R^2 &= 80\% \\ y &= \beta_0 + \beta_1 x_1 + \beta_2 x_2, & R^2 &= 81\% \end{aligned}$$

Is the second model necessarily better than the first one?

Adjusted R^2

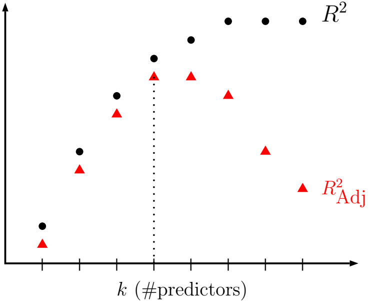
The Adjusted R^2 criterion is more suitable for such comparisons:

$$R^2_{\text{adj}} = 1 - \frac{SS_{Res}/(n-p)}{SS_T/(n-1)}$$

where

- $n-p$: degrees of freedom of SS_{Res}
- $n-1$: degrees of freedom of SS_T

It takes into consideration both the fitting error SS_{Res} and the model size p.



The larger R^2_{adj}, the better the model.

Remark.

- As $p = k + 1$ (model complexity) increases, $n - p$ decreases. SS_{Res} will either decrease or stay the same:
    - If SS_{Res} does not change or decreases by very little, then R^2_{Adj} will decrease. $\longleftarrow$ The smaller model is better
    - If SS_{Res} decreases relatively more than $n - p$ does, then R^2_{Adj} would increase. $\longleftarrow$ The larger model is better
- We can write equivalently

$$R^2_{\mathrm{Adj}} = 1 - \frac{n-1}{n-p}(1 - R^2)$$

This implies that $R^2_{\mathrm{Adj}} < R^2$. When n is very large, the two are of little difference.

Assessing predictive power of a model

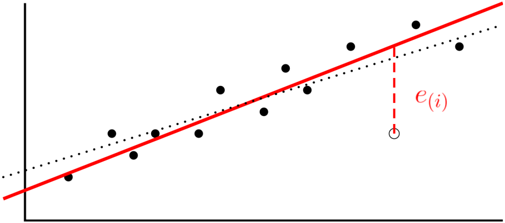
Another way to compare two regression models with different sizes is through their predictive power (using deleted residuals).

Definition

The deleted residuals of a regression model are defined as

$$e_{(i)} = y_i - \hat{y}_{(i)}, \; i = 1, \ldots, n$$

where $\hat{y}_{(i)}$ is the prediction of y_i based on the model fit over all observations except the ith one.



PRESS (prediction sum of squares)

The deleted residuals can be used to define the PRESS statistic for measuring how well a regression model will perform in predicting new data:

$$\mathrm{PRESS} = \sum_{i=1}^{n} e_{(i)}^2$$

Clearly, small values of the PRESS statistic are desired, and they should be looked at relative to SS_T:

$$R^2_{\mathrm{prediction}} = 1 - \frac{\mathrm{PRESS}}{SS_T}$$

Remark. Since $\mathrm{PRESS} > SS_{Res}$, we must have $R^2_{\mathrm{prediction}} < R^2$ in general.

3.3 Other Considerations in the Regression Model

Multicollinearity

A serious issue in multiple linear regression is multicolinearity, or near-linear dependence among the regression variables, e.g., $x_3 \approx 2x_1 + 3x_2$.

- Numerically, the solution of $\mathbf{X}^T\mathbf{X}\hat{\beta} = \mathbf{X}^T\mathbf{y}$ is unstable.
- The redundant predictor contributes no new information about the response:

$$y = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \hat{\beta}_3 x_3 = \hat{\beta}_0 + (\hat{\beta}_1 + 2\hat{\beta}_3)x_1 + (\hat{\beta}_2 + 3\hat{\beta}_3)x_2$$

- The estimated regression coefficients will be arbitrary:

$$y = x_1 + x_2 + 2x_3 = 3x_1 + 4x_2 + x_3 = 5x_1 + 7x_2 = \cdots$$

Such an issue may arise due to too many predictors being collected without noticing their correlation.

Detecting multicollinearity

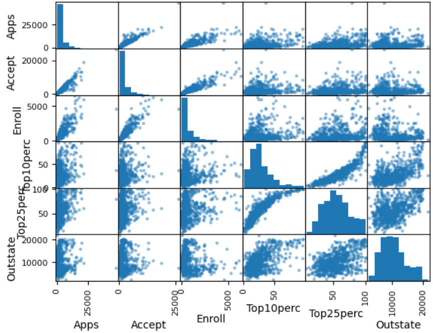
Ideally, we would like to know not only whether there is multicollinearity in the model, but also the severity of the issue (weak, moderate, strong, etc.) and to determine which predictor variable(s) cause the problem.

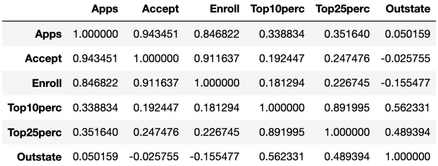
1. **Scatterplot/correlation matrix**: This is a good first step but can only reveal near-linear dependence between a pair of predictors.
2. **Variance inflation factors (VIFs)**: Can detect near-linear dependence among any number of predictors.
3. **Condition number of the correlation matrix**: A large value (1000 or greater) indicates strong multicollinearity in the data.

Detecting correlation between two predictors

When there is a clear linear dependence between two predictors, this can be detected by

- looking at the scatterplot matrix of all predictors ⟵ This can be a bit subjective
- computing the pairwise correlation scores ⟵ better





	Apps	Accept	Enroll	Top10perc	Top25perc	Outstate
Apps	1.000000	0.943451	0.846822	0.338834	0.351640	0.050159
Accept	0.943451	1.000000	0.911637	0.192447	0.247476	-0.025755
Enroll	0.846822	0.911637	1.000000	0.181294	0.226745	-0.155477
Top10perc	0.338834	0.192447	0.181294	1.000000	0.891995	0.562331
Top25perc	0.351640	0.247476	0.226745	0.891995	1.000000	0.489394
Outstate	0.050159	-0.025755	-0.155477	0.562331	0.489394	1.000000

Detecting correlation among three or more predictors

To check for multicollinearity among any number (k) of predictors, we regress each single predictor $x_j,\ j = 1, \ldots, k$ on the remaining ones, i.e.,

$$x_j \ \sim \ x_1 + \cdots + x_{j-1} + x_{j+1} + \cdots + x_k$$

and compute the corresponding coefficients of determination R_j^2.

A large value of R_j^2 indicates strong linear dependence of x_j on the other regressors, thus implying multicollinearity of the predictors in the model.

In practice, we report the Variance Inflation Factors (VIFs) of the predictors instead:

$$\mathrm{VIF}_j = \frac{1}{1 - R_j^2}, \quad j = 1, \ldots, k$$

How to use VIFs to detect multicollinearity

Remark. Consider the following cases:

- When x_j is orthogonal to all the other regressors (ideal case):

$$R_j^2 = 0 \qquad \longrightarrow \qquad \mathrm{VIF}_j = 1$$

- When x_j is nearly a linear combination of the other regressors (bad case):

$$R_j^2 \approx 1 \qquad \longrightarrow \qquad \mathrm{VIF}_j \text{ is large}$$

The larger these factors are, the more you should worry about multicollinearity in your model. A rule of thumb is that if for some j,

$$\mathrm{VIF}_j > 10$$

then multicollinearity is high among the predictors.

How to handle multicollinearity

There are different ways to handle multicollinearity:

- **Variable selection**: select a subset of the variables in some way (e.g., by monitoring the adjusted R^2).
- **Regularization**: Incorporate the model complexity into the objective function in one of the following ways:
  - LASSO
  $$\min_{\hat{\beta}} \|\mathbf{y} - \mathbf{X}\hat{\beta}\|^2 + \lambda \|\hat{\beta}\|_1$$
  - Ridge regression
  $$\min_{\hat{\beta}} \|\mathbf{y} - \mathbf{X}\hat{\beta}\|^2 + \lambda \|\hat{\beta}\|_2^2$$

  where $\lambda \geq 0$ is a tradeoff parameter: large values of λ promote zero values in the components of $\hat{\beta}$.

We will discuss these techniques in detail in Chapter 6.

Feature scaling

The choices of the units of the predictors in a linear model may cause their regression coefficients to have very different magnitudes, e.g.,

$$y = 3 - 20x_1 + 0.01x_2$$

In order to directly compare regression coefficients, we need to standardize the predictors (and sometimes also the response) to be on the same magnitude.

Centering and scaling the predictors also helps reduce the multicollinearity!

Polynomial regression

When a linear model consisting of the original predictors does not seem adequate, for example, when the plot of residuals against fitted values shows curvature, we can add higher order terms to the model such as

- powers of the original features

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{22} x_2^2$$

- or interactions of them,

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{12} x_1 x_2$$

- or even a mixture of powers and interactions of them

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{11} x_1^2 + \beta_{22} x_2^2 + \beta_{12} x_1 x_2$$

This is called polynomial regression: powerful but easy to overfit.

Categorical predictors

Categorical predictors can also be included in the model, in the form of indicator variables.

For example, let x_1 be numerical but x_2 a categorical variable with 3 levels, A, B, C.

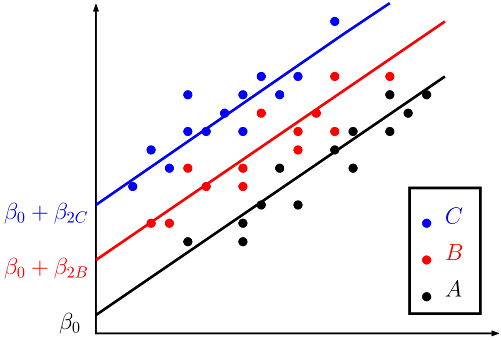
The categorical variable x_2 generates two indicator variables for two of the levels (say B, C) relative to a selected baseline (say A),

$$x_{2B} = 1_{\{x_2=B\}}, \qquad x_{2C} = 1_{\{x_2=C\}},$$

The effective model is

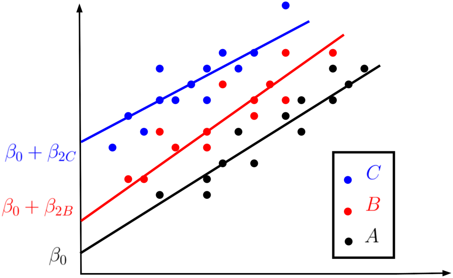
$$y = \beta_0 + \beta_1 x_1 + \beta_{2B} x_{2B} + \beta_{2C} x_{2C} = \begin{cases} \beta_0 + \beta_1 x_1, & x_2 = A \\ (\beta_0 + \beta_{2B}) + \beta_1 x_1, & x_2 = B \\ (\beta_0 + \beta_{2C}) + \beta_1 x_1, & x_2 = C \end{cases}$$

x_2	x_{2B}	x_{2C}
A	0	0
A	0	0
A	0	0
B	1	0
B	1	0
B	1	0
B	1	0
C	0	1
C	0	1



It is also possible to fit nonparallel lines to the different levels of x_2 by adding interaction terms between the indicator variables x_{2B}, x_{2C} and the continuous feature x_1:

$$
\begin{aligned}
y &= \beta_0 + \beta_1 x_1 + \beta_{2B} x_{2B} + \beta_{2C} x_{2C} + \beta_{12B} x_1 x_{2B} + \beta_{12C} x_1 x_{2C} \\
&= \begin{cases} \beta_0 + \beta_1 x_1, & x_2 = A \\ (\beta_0 + \beta_{2B}) + (\beta_1 + \beta_{12B}) x_1, & x_2 = B \\ (\beta_0 + \beta_{2C}) + (\beta_1 + \beta_{12C}) x_1, & x_2 = C \end{cases}
\end{aligned}
$$



Optional: Linear regression via gradient descent

Computing via gradient descent

The multiple linear regression problem, when $\mathbf{X} \in \mathbb{R}^{n\times d}$ has full column rank, has an analytic solution:

$$\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}.$$

On one hand, it is very nice; on the other hand, it may be computationally expensive when both n, d are large:

- multiplying $\mathbf{X}^T$ and $\mathbf{X}$ together: $\mathcal{O}(nd^2)$ complexity
- inverting $\mathbf{X}^T\mathbf{X}$: $\mathcal{O}(d^3)$ complexity.

It is possible to use the matrix SVD to compute $(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$ altogether more efficiently.

Here, we present a gradient-based numerical approach to finding the least squares regression coefficients $\hat{\beta}$.

More generally, let us consider the unconstrained minimization problem

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x})$$

Suppose the objective function $f(\mathbf{x})$ is differentiable everywhere on $\mathbb{R}^n$.

At an arbitrary point $\mathbf{x}_0 \in \mathbb{R}^n$, we know that the negative gradient $-\nabla f(\mathbf{x}_0)$ represents the direction of fastest decrease for the function f.

For example, if $f(\mathbf{x}) = x_1^2 + x_2^2$, then

$$-\nabla f = -(2x_1, 2x_2).$$

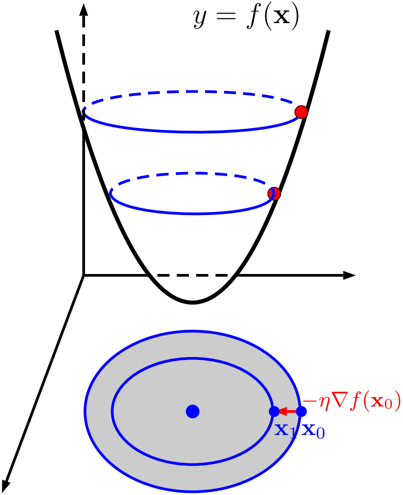
Thus, if we move a small step away from $\mathbf{x}_0$ along that direction, i.e.,

$$\mathbf{x}_1 = \mathbf{x}_0 - \eta \cdot \nabla f(\mathbf{x}_0)$$

then we can get to a new point $\mathbf{x}_1$ where the value of f is smaller :

$$\begin{aligned} f(\mathbf{x}_1) - f(\mathbf{x}_0) &\approx \nabla f(\mathbf{x}_0)^T \cdot (\mathbf{x}_1 - \mathbf{x}_0) \\ &= -\eta \|\nabla f(\mathbf{x}_0)\|^2 < 0. \end{aligned}$$

Here, $\eta > 0$ is called the *learning rate*, and its value should be properly set by the user (not too large, not too small).



Now, if we repeat the above step over and over at the new locations to generate a sequence of points, $\{\mathbf{x}_t\}_{t\geq 0} \subset \mathbb{R}^n$, i.e.,

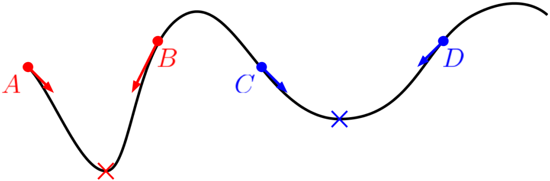
$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta \cdot \nabla f(\mathbf{x}_t), \quad t = 0, 1, 2, \ldots$$

we expect the corresponding values of the objective function, $\{f(\mathbf{x}_t)\}_{t\geq 0}$, to continuously decrease until convergence is reached (which is when the process is stopped).

This process is called *gradient descent*, and the above iteration formula is called the *update rule*.

However, it should be noted that gradient descent only converges to a local minimum of the objective function, which may or may not be the global minimum.

When the function has more than one local minimum (and one of them will be the global minimum), gradient descent can converge to any of them, depending on the starting location.



In many machine learning problems (such as neural networks), convergence to a local minimum is still useful because of their vast complexity.

We explain how well gradient descent works in multiple linear regression.

First, to help with parameter tuning, we change the objective function to

$$S(\hat{\beta}) = \frac{1}{2n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \frac{1}{2n}\|\mathbf{y} - \mathbf{X}\hat{\beta}\|^2$$

Using the gradient formula we have obtained earlier,

$$\frac{\partial S}{\partial \hat{\beta}} = \frac{1}{2n}\left(2\mathbf{X}^T\mathbf{X}\hat{\beta} - 2\mathbf{X}^T\mathbf{y}\right) = \frac{1}{n}\mathbf{X}^T\left(\mathbf{X}\hat{\beta} - \mathbf{y}\right)$$

we can write down the following iteration scheme for gradient descent

$$\hat{\beta}_{t+1} = \hat{\beta}_t - \frac{\eta}{n}\mathbf{X}^T\left(\mathbf{X}\hat{\beta}_t - \mathbf{y}\right), \quad i \geq 0$$

where both $\hat{\beta}_0$ and η need to specified by the user.

How to set the learning rate:

- The learning rate η cannot be too large (fail to converge), or too small (converge very slowly).
- One can monitor the value of the objective function,

$$S(\hat{\beta}) = \frac{1}{2n}\|\mathbf{y} - \mathbf{X}\hat{\beta}\|^2$$

  to determine if η is too small or too large.

Comparing with the exact method, the gradient descent has a huge reduction in computational complexity because the update rule is based only on matrix vector multiplications.

The actual complexity in each iteration requires $\mathcal{O}(nd)$ time, while the exact method requires $\mathcal{O}(d^2(n+d))$ time.

Since the objective function of the least square problem is convex, gradient descent is guaranteed to converge to the global minimum.

How much time it takes gradient descent to find the global minimum depends on both the initial point $\hat{\beta}_0$ and the learning rate η.

Stochastic gradient descent

When $\mathbf{X}$ is extremely large, gradient descent can also be slow.

In such cases, one can divide the data into many small subsets of equal size m (called mini batches), say $(\mathbf{X}_1, \mathbf{y}_1), (\mathbf{X}_2, \mathbf{y}_2), \ldots$ and use them, one at a time, to perform the updates:

```
for $t = 1, 2, \ldots, T$ ($T$ is the total number of epochs)
    for $i = 1, 2, \ldots, n/m$,
        $\hat{\boldsymbol{\beta}}_t \longleftarrow \hat{\boldsymbol{\beta}}_t - \frac{\eta}{m}\mathbf{X}_i^T\left(\mathbf{X}_i\hat{\boldsymbol{\beta}}_t - \mathbf{y}_i\right), \quad i \geq 0$
```