Lecture 7: QR decomposition and least square models

Francesco Preta
07/28/2020

Orthogonal projection

Definition

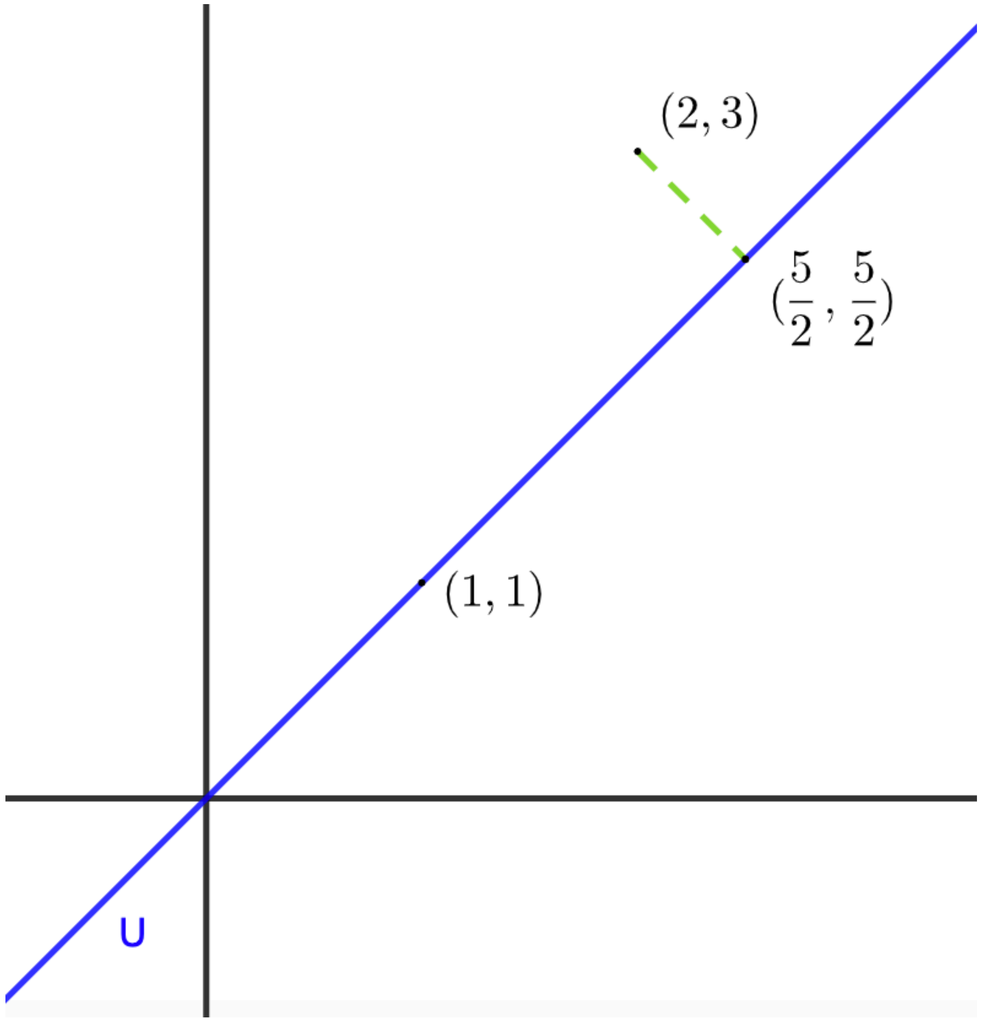
Let U be a subspace of $\mathbb{R}^n$. For $\mathbf{y} \in \mathbb{R}^n$, we define the orthogonal projection $\text{proj}_U(\mathbf{y})$ as

$$\text{proj}_U(\mathbf{y}) = \text{argmin}_{\hat{\mathbf{y}} \in U} ||\mathbf{y} - \hat{\mathbf{y}}||^2$$

Orthogonal projection in $\mathbb{R}^2$

Let $U = \text{span}(\begin{bmatrix} 1 \\ 1 \end{bmatrix})$ and $\mathbf{y} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$

Orthogonal projection in $\mathbb{R}^2$ (continued)

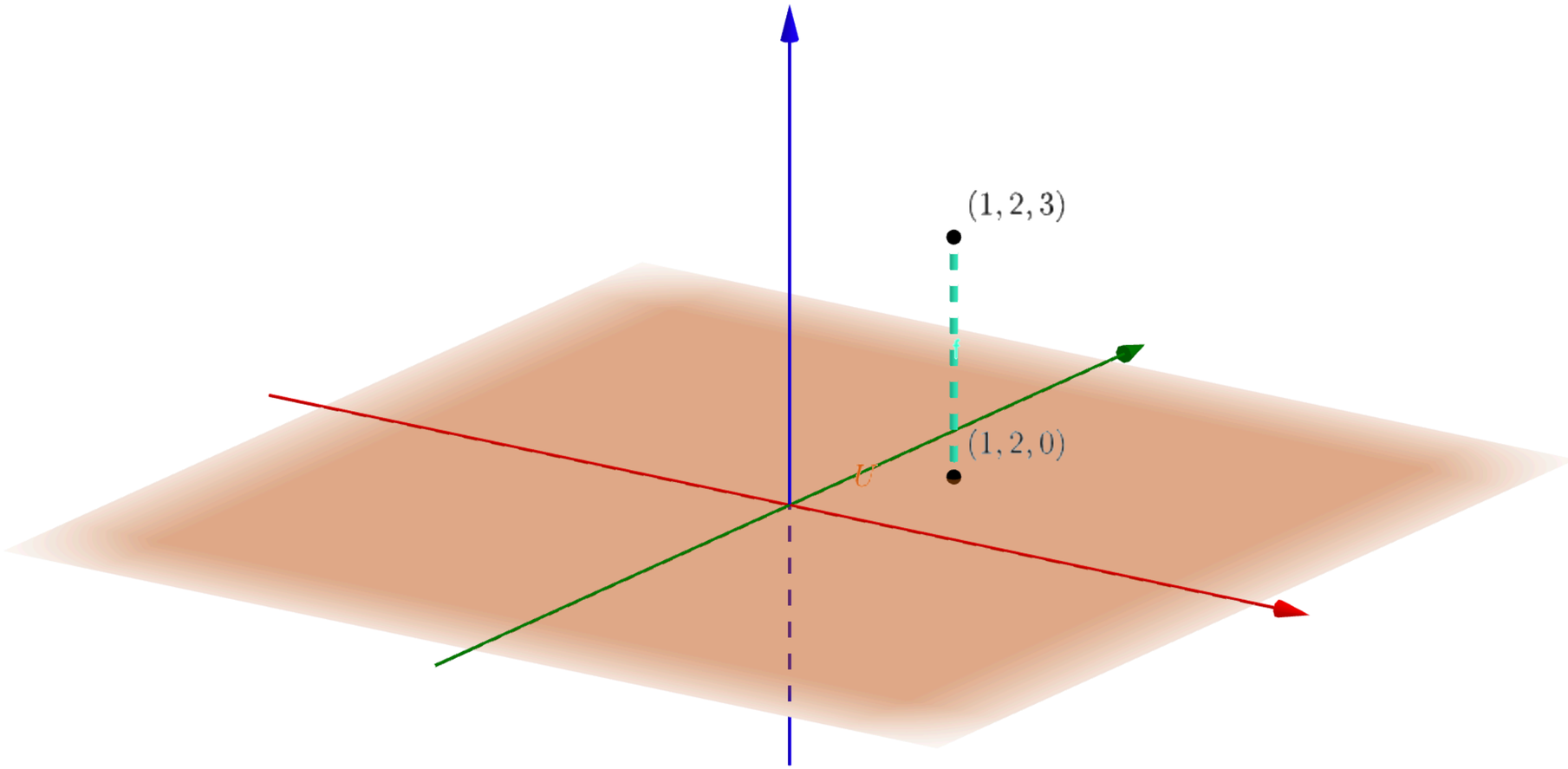


Orthogonal projection in $\mathbb{R}^3$

$$U = \text{span}(\mathbf{e}_1, \mathbf{e}_2) = \left\{ \begin{bmatrix} \alpha \\ \beta \\ 0 \end{bmatrix} \mid \alpha, \beta \in \mathbb{R} \right\}$$

$$\mathbf{y} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

Orthogonal projection in $\mathbb{R}^3$ (continued)



General method to find orthogonal projection

Theorem

Let U be a subspace of $\mathbb{R}^n$ of dimension k and let $\mathcal{U} = \{\mathbf{u}_i\}_{i=1}^k$ be an orthonormal basis for U. Then for every $\mathbf{y} \in \mathbb{R}^n$,

$$\text{proj}_U \mathbf{y} = \sum_{i=1}^{k} \langle \mathbf{y}, \mathbf{u}_i \rangle \mathbf{u}_i$$

Example:

$$\mathbf{u}_1 = \begin{bmatrix} \frac{1}{\sqrt{3}} \\ 0 \\ \frac{1}{\sqrt{3}} \\ -\frac{1}{\sqrt{3}} \end{bmatrix} \quad \mathbf{u}_2 = \begin{bmatrix} \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{3}} \\ -\frac{1}{\sqrt{3}} \\ 0 \end{bmatrix} \quad \mathbf{u}_3 = \begin{bmatrix} -\frac{2}{\sqrt{15}} \\ \frac{1}{\sqrt{15}} \\ -\frac{1}{\sqrt{15}} \\ -\frac{3}{\sqrt{15}} \end{bmatrix}, \mathbf{y} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Orthogonal complement

$$U^{\perp} = \{\mathbf{v} \in \mathbb{R}^n \mid \langle \mathbf{v}, \mathbf{u} \rangle = 0 \text{ for every } \mathbf{u} \in U\}$$

Direct sum decomposition

Theorem

Let U be a subspace of $\mathbb{R}^n$, then for every $\mathbf{y} \in \mathbb{R}^n$ there exists a unique decomposition

$$\mathbf{y} = \text{proj}_U \mathbf{y} + \text{proj}_{U^\perp} \mathbf{y} \, .$$

In particular, this means that

$\text{proj}_{U^\perp} \mathbf{y} = \mathbf{y} - \text{proj}_U \mathbf{y}$.

In the previous example:

$$\mathbf{u}_1 = \begin{bmatrix} \frac{1}{\sqrt{3}} \\ 0 \\ \frac{1}{\sqrt{3}} \\ -\frac{1}{\sqrt{3}} \end{bmatrix} \quad \mathbf{u}_2 = \begin{bmatrix} \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{3}} \\ -\frac{1}{\sqrt{3}} \\ 0 \end{bmatrix} \quad \mathbf{u}_3 = \begin{bmatrix} -\frac{2}{\sqrt{15}} \\ \frac{1}{\sqrt{15}} \\ -\frac{1}{\sqrt{15}} \\ -\frac{3}{\sqrt{15}} \end{bmatrix}, \mathbf{y} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Orthonormal basis decomposition

If $\{\mathbf{u}_i\}_{i=1}^n$ is an orthonormal basis of $\mathbb{R}^n$, we have that for every $\mathbf{y} \in \mathbb{R}^n$, we can write

$$\mathbf{y} = \sum_{i=1}^{n} \langle \mathbf{y}, \mathbf{u}_i \rangle \mathbf{u}_i \, .$$

Proof:

QR Decomposition

Decomposing a full-rank matrix

Consider $\{\mathbf{b}_i\}_{i=1}^k$ the columns of a full rank $n \times k$ matrix B, for $k \leq n$. If $\{\mathbf{u}_i\}_{i=1}^k$ are the orthonormal vectors obtained by applying the Gram-Schmidt algorithm, thenspan$(\{\mathbf{b}_i\}_{i=1}^k) =$ span$(\{\mathbf{u}_i\}_{i=1}^k)$. So we can write

$$\mathbf{b}_j = \sum_{i=1}^{k} \langle \mathbf{b}_j, \mathbf{u}_i \rangle \mathbf{u}_i \, .$$

However, $U_j =$ span$(\mathbf{u}_1, \ldots, \mathbf{u}_j) =$ span$(\mathbf{b}_1, \ldots, \mathbf{b}_j)$ and $\mathbf{u}_l \in U_j^{\perp}$ for $l > j$.

Therefore $\langle \mathbf{b}_j, \mathbf{u}_l \rangle = 0$ for $l > j$ and we can write

$$\mathbf{b}_j = \sum_{i=1}^{j} \langle \mathbf{b}_j, \mathbf{u}_i \rangle \mathbf{u}_i$$

QR Decomposition

Now consider the following matrices:

$$Q = [\quad \mathbf{u}_1 \quad \mathbf{u}_2 \quad \dots \quad \mathbf{u}_k]$$

$$R = \begin{bmatrix} \langle \mathbf{b}_1, \mathbf{u}_1 \rangle & \langle \mathbf{b}_2, \mathbf{u}_1 \rangle & \dots & \langle \mathbf{b}_k, \mathbf{u}_1 \rangle \\ 0 & \langle \mathbf{b}_2, \mathbf{u}_2 \rangle & \dots & \langle \mathbf{b}_k, \mathbf{u}_1 \rangle \\ \dots & & & \\ 0 & 0 & \dots & \langle \mathbf{b}_k, \mathbf{u}_k \rangle \end{bmatrix}$$

Then the QR decomposition of B is given by $B = QR$.

Dimension of the QR decomposition

If B is not a square matrix, then Q is a $n \times k$ matrix and R is a $k \times k$ matrix. In general R is upper triangular and if Q is square, then it is orthogonal, that is $Q^T Q = I$

Example:

$$C = \begin{bmatrix} 0 & 1 & -1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

Solving square systems of linear equations

Consider the square system of linear equation $A\mathbf{x} = \mathbf{b}$ with A invertible. For $A = QR$, since Q is orthogonal, we have that $QR\mathbf{x} = \mathbf{b}$ is equivalent to solving $R\mathbf{x} = Q^T\mathbf{b}$. However, we know that R is upper triangular, so the system can be solved using back-substitution.

Example:

$$\begin{cases} x_1 + x_2 = 1 \\ -x_1 + x_2 + x_3 = -1 \\ x_1 + x_3 = 1 \end{cases}$$

Computing the inverse

Let A be an invertible $n \times n$ matrix. Since

$$A^{-1} = R^{-1}Q^{-1} = R^{-1}Q^{T}$$

in order to find the inverse matrix, one can solve a system $RB = Q^{T}$, or equivalently, for $\mathbf{b}_i$ the i-th column of B and for $\tilde{\mathbf{q}}_i$ the i-th column of Q^{T}, solve the system

$$R\mathbf{b}_i = \tilde{\mathbf{q}}_i \qquad \text{for } i = 1,\ldots,n$$

Since R is a triangular matrix, this can be solved by backsubstitution.

Example:

$$A = \begin{bmatrix} 1 & 1 & 0 \\ -1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

Computational complexity

The QR decomposition for a $n \times k$ matrix (for $n \geq k$) requires a number of flops depending on the Gram-Schmidt algorithm. For every vector $\mathbf{b}_i$, for $i = 1,...,k$, we need to compute i scalar products, $i$scalar multiplications and $i - 1$ sums, then the normalization takes approximately $2n$ flops, so that

$$tot = \sum_{i=1}^{k} [(2n-1)i + ni + n(i-1) + 2n] = nk + (4n-1)\sum_{i=1}^{k} i =$$

$$nk - k + (4n-1)\frac{k(k+1)}{2}$$

Least square models

Assumptions of the models

Consider the equation $A\mathbf{x} = \mathbf{b}$, for known $A \in \mathbb{R}^{n \times k}$ and $\mathbf{b} \in \mathbb{R}^n$ and suppose that:

- $\mathbf{b}$ is not in the span of the columns of A.
- The columns of A are linearly independent.

This means that the system is over-determined ($n > k$) and that it does not admit a solution.

Least square solutions

It makes sense to find a solution $\mathbf{z}$ which, in some sense, satisfies the conditions with a small margin of error. In other words, we can choose $\mathbf{z}$ which minimizes the sum of squares, that is

$$\mathbf{z} = \text{argmin}_{\mathbf{x}\in\mathbb{R}^k} ||\mathbf{b} - A\mathbf{x}||^2$$

Since we are minimizing the square of the distance between $\mathbf{b}$ and $A\mathbf{x}$, this is called a **least square method** and $\mathbf{z}$ is called the **least square solution** of the system.

Analytic solution

The problem can be solved analytically using calculus to minimize the function $lsq(\mathbf{x}) = ||\mathbf{b} - A\mathbf{x}||^2$. Then

$$\frac{\partial lsq}{\partial x_k} = -2\sum_{i=1}^{m}(\mathbf{b}_i - \sum_{j=1}^{n} A_{i,j}x_j)A_{i,k}$$

Solution of the system

$\mathbf{z}$ is the solution of a system of linear equations

$$\sum_{j=1}^{n}\sum_{i=1}^{m} A_{k,i}^{T} A_{i,j} z_j = \sum_{i=1}^{m} A_{k,i}^{T}\mathbf{b}_i \qquad k = 1,..,n$$

which is a system of the type

$$A^T A\mathbf{z} = A^T\mathbf{b}$$

Explicit form of the solution

A^TA is a $k \times k$ invertible matrix, since A has rank k. The solution is given by

$$\mathbf{z} = (A^TA)^{-1}A^T\mathbf{b}$$

We have already seen that the matrix $(A^TA)^{-1}A^T$ is called the pseudo-inverse of A and is denoted $A^\dagger$. Its existence is guaranteed by the fact that A is a full-rank tall matrix.

Solving least square models with QR decomposition

If A is a full rank $n \times k$ tall matrix, then we can find a QR decomposition $A = QR$. Then the pseudo-inverse is given by

$$A^{\dagger} = (A^T A)^{-1} A^T = (R^T Q^T Q R)^{-1} R^T Q^T = R^{-1} (R^T)^{-1} R^T Q^T = R^{-1} Q^T$$

Finding the pseudo-inverse

To find the pseudo-inverse it therefore suffices to solve the system of linear equations $RA^{\dagger} = Q^{T}$, that reduces to $R\tilde{\mathbf{a}}_i = \tilde{\mathbf{q}}_i$ for $i = 1,...,m$, which can still be solved through back-substitution.

Finding the solution to the least square problem

The least square solution to the general linear system $\mathbf{z} = A^{\dagger}\mathbf{b}$ can also be found through QR decomposition, for which it's sufficient to solve by back-substitution $R\mathbf{z} = Q^{T}\mathbf{b}$.

Example:

$$\begin{cases} x_2 - x_3 = 2 \\ x_1 + 2x_3 = 3 \\ x_2 + x_3 = 2 \\ x_1 - 2x_3 = 0 \end{cases}$$

Geometric interpretation of least square problem

Geometrically, the problem can be treated as finding a linear combination of the columns of A that minimizes the distance from a given point $\mathbf{b}$, or equivalently, finding $\mathbf{z}$ such that $A\mathbf{z} = \text{proj}_V \mathbf{b}$ where $V = \text{span}(\mathbf{a}_1, \ldots, \mathbf{a}_k)$.

Least square regression

In statistics, least square regression is widely used to understand correlation between variables. In the simplest regression model, two numerical variables are compared to see if one can express correlation between them. A priori, a decision is made on which variable is the **independent variable** x and which one is regressed onto the other.

Once that decision is made, one needs to find the parameters (β, α) that best describe the relationship

$$y = \beta x + \alpha$$

Least square regression (continued)

The data is given by a set of observations in pair $\{(\hat{x}_i, \hat{y}_i)\}_{i=1}^{N}$ for which one can find β and α that minimize the sum of the squared errors between each observation $\hat{y}_i$ and the predicted quantity $y_i = \beta\hat{x}_i + \alpha$:

$$\min_{\alpha,\beta} \sum_{i=1}^{N} (\hat{y}_i - y_i)^2$$

for this reason this is called **least square regression**.

Analytic solution

We want to find α and β that minimize the function

$$lsq(\alpha, \beta) = \sum_{i=1}^{N} (\hat{y}_i - \beta \hat{x}_i - \alpha)^2$$

Mean, variance and covariance

We define the following quantities:

- **Mean**: $\bar{x} = \frac{1}{N}\sum_{i=1}^{N} \hat{x}_i$.
- **Covariance** between x and y: $COV(x, y) = \frac{1}{N}\sum_{i=1}^{N} (\hat{y}_i - \bar{y})(\hat{x}_i - \bar{x})$
- **Variance** of x: $Var(x) = COV(x, x)$

We have the following properties:

- $$COV(x, y) = \frac{1}{N}\sum_{i=1}^{N} \hat{y}_i\hat{x}_i - \bar{x}\bar{y}$$
- $$Var(x) = \frac{1}{N}\sum_{i=1}^{N} \hat{x}_i^2 - \bar{x}^2$$

Solution of the regression

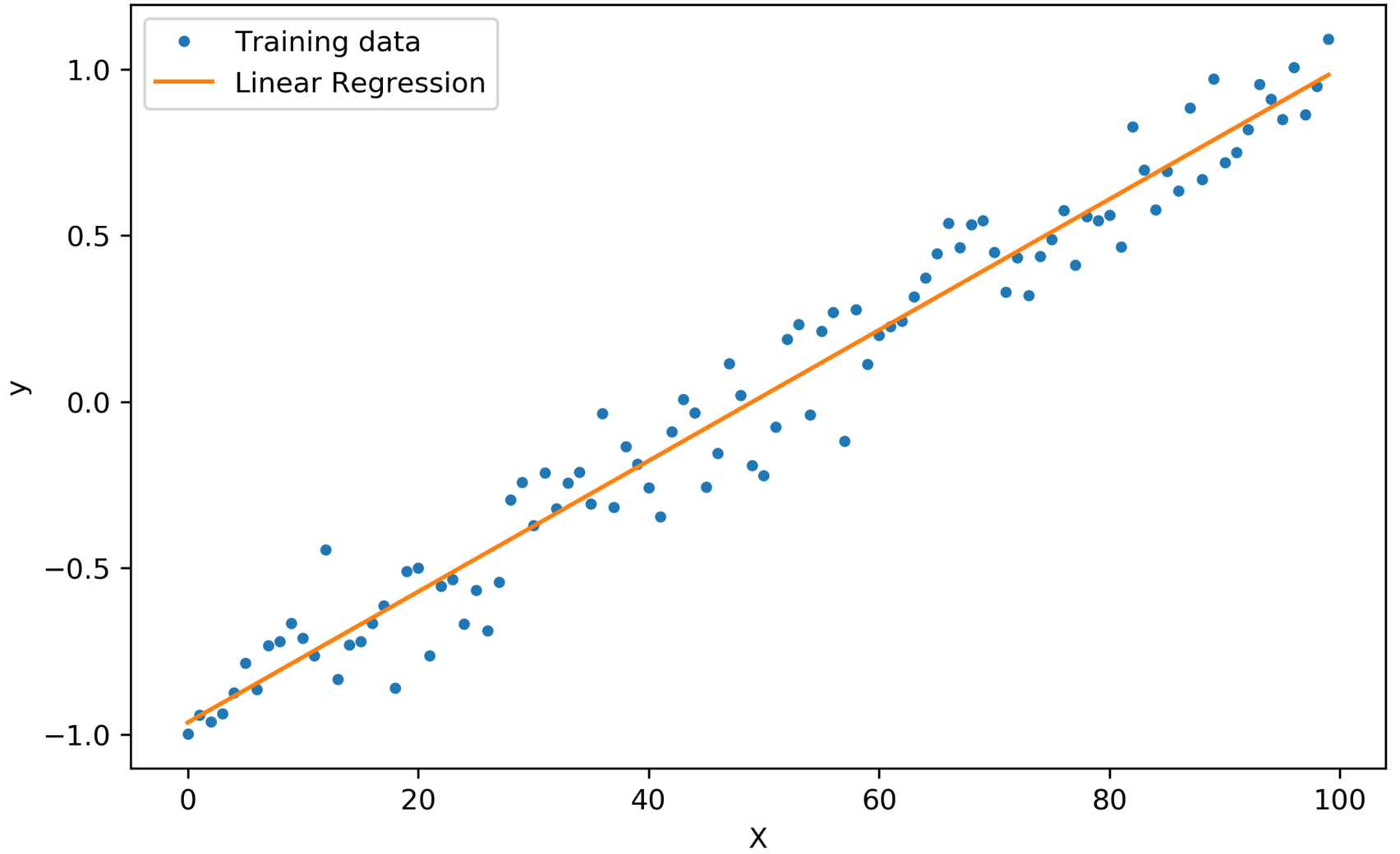
$$\beta = \frac{COV(x, y)}{Var(x)}$$

$$\alpha = \bar{y} - \beta\bar{x}$$

Geometric intuition

The geometric meaning of the linear regression is the following: we try to find the line $y = \beta x + \alpha$ that best approximates the data points $\{(\hat{x}_i, \hat{y}_i)\}_{i=1}^{N}$ in $\mathbb{R}^2$.

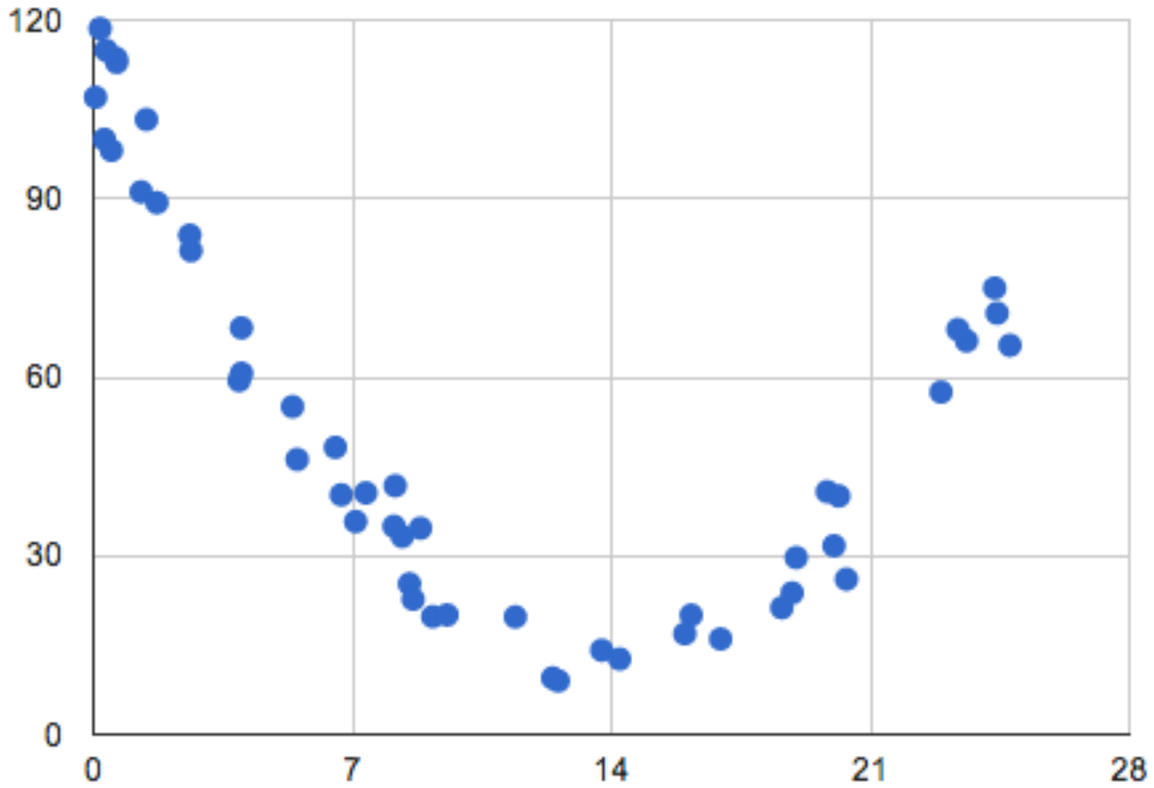
Geometric intuition (continued)



Failure of linear regression

In some situation, the regression line is successful in capturing the relation between two variables. However, this is not always the case, since the relation can be captured by a non-linear model.

Failure of linear regression (continued)



Coefficient of determination

A good way to measure whether the regression model successfully captures the relation between the variables is given by the coefficient of determination R^2, defined as

$$R^2 = 1 - \frac{\sum_{i=1}^{N} (\hat{y}_i - y_i)^2}{\sum_{i=1}^{N} (\hat{y}_i - \bar{y})^2}$$

Coefficient of determination (continued)

The denominator in the formula constitutes the variance of y, which explains the general variation from the average value, while the numerator is given by the sum of squares. The R^2 is meant to explain how much of the total variation is captured by the model. A low sum of squares with respect to the total variation, will give a high value of R^2, while a high such sum will give a low value for R^2, thus showing that the model fails to capture most of the variation.

Linear regression and least square models

When it comes to relating the linear regression model to our general linear least square method, notice that we can choose $\mathbf{b} = \mathbf{y}$ as a N-dimensional vector such that $\mathbf{y}_i = \hat{y}_i$ and A as a $N \times 2$ matrix having $A_{i,1} = \hat{x}_i$ and $A_{i,2} = 1$. Then finding α, β corresponds to finding the orthogonal projection of $\mathbf{y} = \mathbf{b}$ in span$(\mathbf{x}, \mathbf{1})$.

Multilinear regression

More generally, we can consider linear methods in which the dependent variable $y$depends on k features $(x^1, \ldots, x^k)$, for which we have a set of N observations $\{(\hat{x}_i^1, \ldots, \hat{x}_i^k, \hat{y}_i)\}_{i=1}^N$. Then the least square regression finds the $k+1$ coefficients $(\beta_i, \ldots, \beta_k, \alpha)$ that minimize the sum of the distances between the observation $\hat{y}_i$ and the predicted value $y_i = \sum_{i=1}^k \beta_k x_i^k + \alpha$, that is

$$(\beta_1, \ldots, \beta_k, \alpha) = \text{argmin}_{\beta,\alpha} \sum_{i=1}^{N} ||\hat{y}_i - y_i||^2$$

Multilinear regression as a least square model

This can also be interpreted as a particular instance of our general linear model, in the case in which we are trying to find the linear combinations of the columns of a $N \times (k+1)$ matrix A, with $A_{i,j} = \hat{x}_i^j$ for $j = 1,...,k$ and $A_{i,k+1} = 1$ for $i = 1,...,N$. Then the above problem corresponds to finding

$\text{argmin}_{\mathbf{z} \in \mathbb{R}^{k+1}} ||A\mathbf{z} - \mathbf{y}||$.