

# Linear Regression

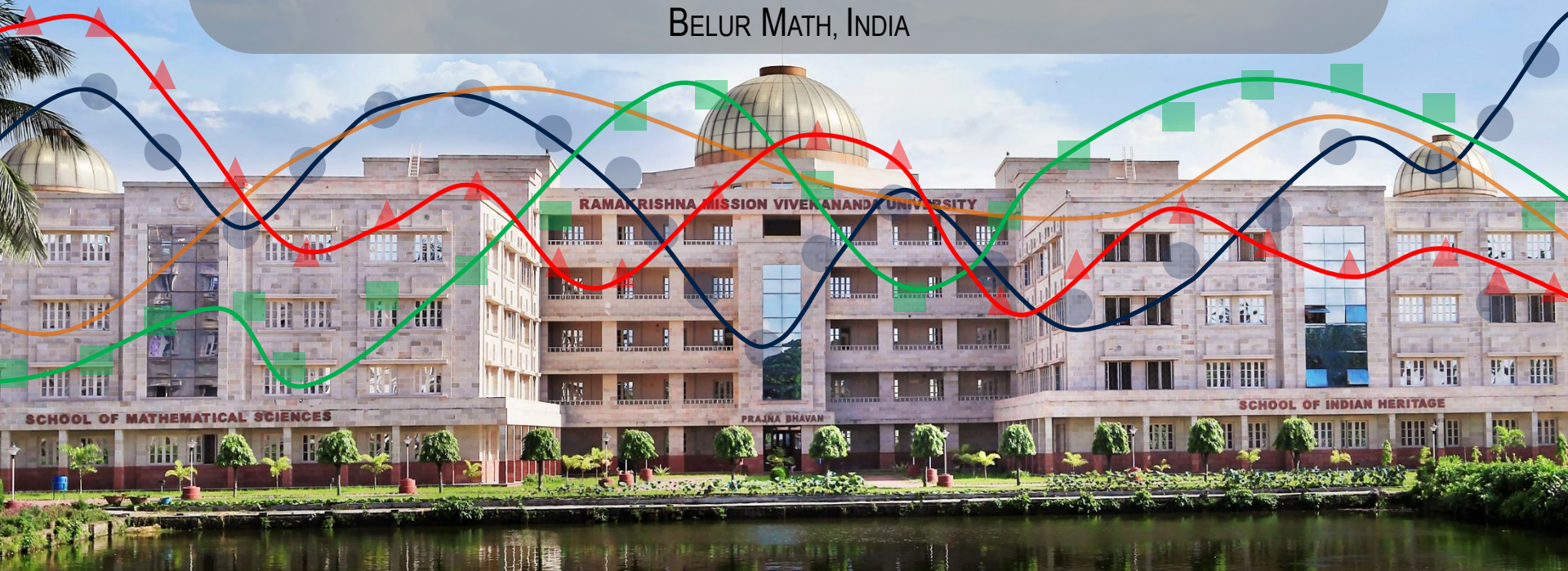
---

**DRIPTA MJ**

Department of Mathematics

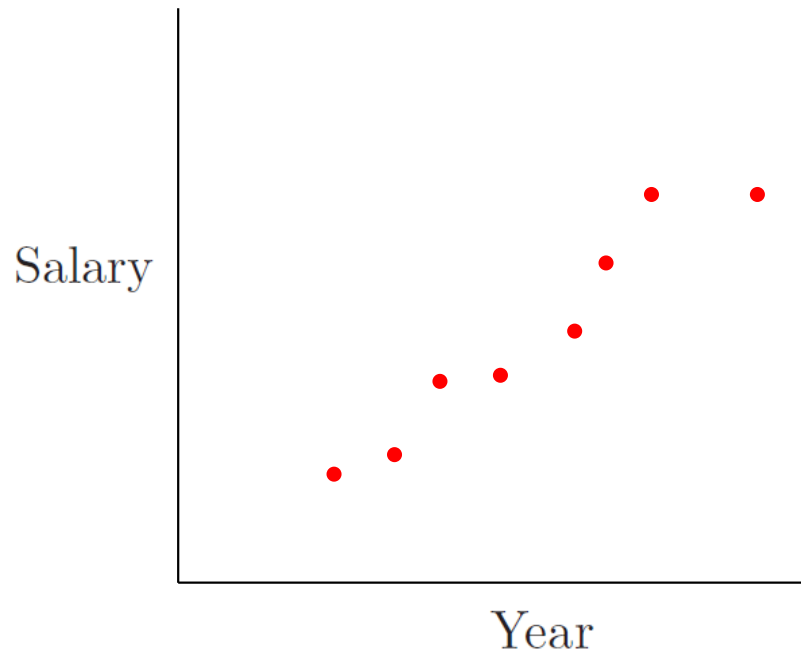
RAMAKRISHNA MISSION VIVEKANANDA EDUCATIONAL AND RESEARCH INSTITUTE

BELUR MATH, INDIA



# Introduction

- Variation of salary with time:



# Notation

- Given a data set of  $N$  points.
- Input data comprise  $D$  features, suppose they are  $x_1, x_2, \dots, x_D$ .
- Representation of the  $i$ th input data point:

$$\mathbf{x}^{(i)} = [x_1^{(i)}, x_2^{(i)}, \dots, x_D^{(i)}]^T$$

- Set of input data points:

$$\mathbf{X} = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(N)}\}$$

- Set of outputs for the  $N$  data points:

$$\mathbf{y} = \{y^{(1)}, y^{(2)}, \dots, y^{(N)}\}$$

# Linear Regression

- Input variables can be written in vectorial form as:

$$\mathbf{x} = [x_1, x_2, \dots, x_D]^T$$

- Prediction: Linear combination of input variables

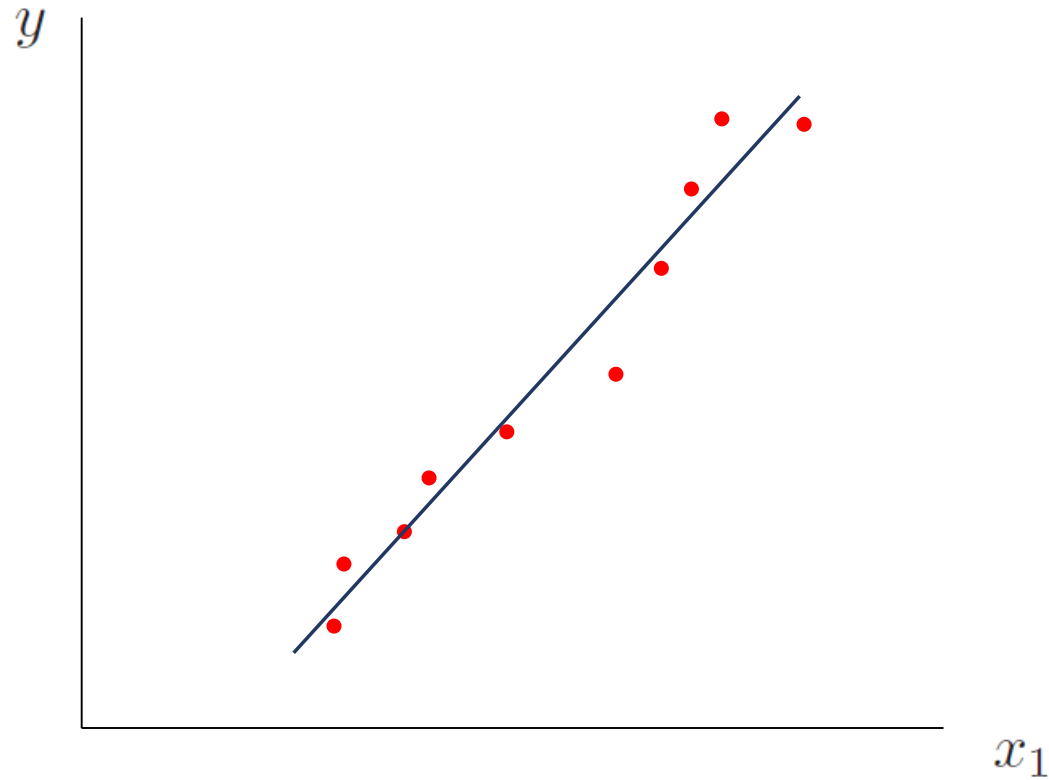
$$z(\mathbf{x}, p_0, p_1, \dots, p_D) = p_0 + p_1 x_1 + p_2 x_2 + \dots + p_D x_D$$

- Vectorial representation of weights:

$$\mathbf{p} = [p_0, p_1, p_2, \dots, p_D]^T$$

# Simplest case

- One input variable:  $x_1$



$$z = p_0 + p_1 x_1$$

# Using basis functions

- Output: Linear combination of functions of input variables

$$z(\mathbf{x}, p) = p_0 + \sum_{i=1}^Q p_i f_i(\mathbf{x})$$

where  $f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_Q(\mathbf{x})$  are the basis functions.

- Taking  $f_0 = 1$ , can write

$$z(\mathbf{x}, p) = \sum_{i=0}^Q p_i f_i(\mathbf{x})$$

- Vectorial representation:

$$z(\mathbf{x}, \mathbf{p}) = \mathbf{p}^T \mathbf{f}(\mathbf{x})$$

where  $\mathbf{p} = [p_0, p_1, \dots, p_Q]^T$  and  $\mathbf{f}(\mathbf{x}) = [f_0(\mathbf{x}), f_1(\mathbf{x}), \dots, f_Q(\mathbf{x})]^T$

# Basis functions

- Polynomial basis functions (with single input  $x_1$ ) have the form:

$$f_0(x_1) = 1, \quad f_1(x_1) = x_1, \quad f_2(x_1) = x_1^2, \quad \dots \quad f_Q(x_1) = x_1^Q,$$

and therefore

$$\mathbf{f}(x_1) = [1, x_1, x_1^2, \dots, x_1^Q]^T.$$

- More complex functions can be generated by increasing the degree of the polynomial  $Q$ .
- Some other popular choices of basis functions:
  - Spline functions (piecewise polynomial curves)
  - Gaussian basis functions

# Error

- Dataset comprise  $N$  data points.
- Outputs at the training locations:  $\mathbf{y} = \{y^{(1)}, y^{(2)}, \dots, y^{(N)}\}$ .
- Predictions at the training locations:  $\mathbf{z} = \{z^{(1)}, z^{(2)}, \dots, z^{(N)}\}$ .
- Sum of squared errors:

$$\begin{aligned}\text{SE} &= \sum_{n=1}^N (y^{(n)} - z^{(n)})^2 \\ &= \sum_{n=1}^N \left( y^{(n)} - \mathbf{p}^T \mathbf{f}(\mathbf{x}^{(n)}) \right)^2\end{aligned}$$

- Mean squared error  $\text{MSE} = \frac{\text{SE}}{N}$
- Often the error function is written in the following form that is more amenable to differentiation

$$E = \frac{1}{2} \sum_{n=1}^N \left( y^{(n)} - \mathbf{p}^T \mathbf{f}(\mathbf{x}^{(n)}) \right)^2$$



# Parameter values

- Optimal weights – Setting the gradient of  $E(\mathbf{p})$  to zero:

$$\nabla E(\mathbf{p}) = 0$$

which yields

$$\sum_{n=1}^N y^{(n)} \mathbf{f}(\mathbf{x}^{(n)})^T - \mathbf{p}^T \left( \sum_{n=1}^N \mathbf{f}(\mathbf{x}^{(n)}) \mathbf{f}(\mathbf{x}^{(n)})^T \right) = 0$$

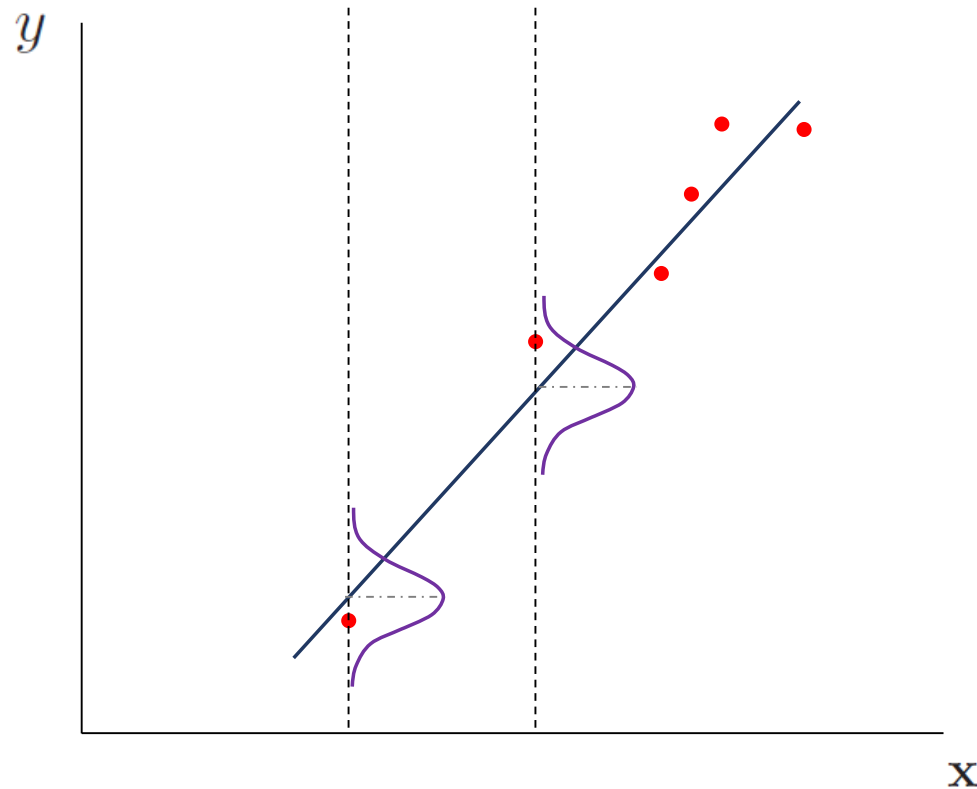
Let

$$\mathcal{F} = \begin{bmatrix} \mathbf{f}(\mathbf{x}^{(1)})^T \\ \mathbf{f}(\mathbf{x}^{(2)})^T \\ \vdots \\ \mathbf{f}(\mathbf{x}^{(N)})^T \end{bmatrix} = \begin{bmatrix} f_0(\mathbf{x}^{(1)}) & f_1(\mathbf{x}^{(1)}) & \cdots & f_Q(\mathbf{x}^{(1)}) \\ f_0(\mathbf{x}^{(2)}) & f_1(\mathbf{x}^{(2)}) & \cdots & f_Q(\mathbf{x}^{(2)}) \\ \vdots & \vdots & \cdots & \vdots \\ f_0(\mathbf{x}^{(N)}) & f_1(\mathbf{x}^{(N)}) & \cdots & f_Q(\mathbf{x}^{(N)}) \end{bmatrix}$$

- Finally, the value of the weights  $\mathbf{p}$  are obtained as:

$$\mathbf{p} = (\mathcal{F}^T \mathcal{F})^{-1} \mathcal{F}^T \mathbf{y}$$

# Probabilistic approach



- Assumption: Gaussian noise model

$$y^{(n)} \sim \mathcal{N}(\mathbf{p}^T \mathbf{x}^{(n)}, \sigma^2)$$

# Maximum likelihood estimator

- Training data comprise  $N$  data points:

$$\mathbf{X} = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(N)}\}, \quad \mathbf{y} = \{y^{(1)}, y^{(2)}, \dots, y^{(N)}\}$$

- Likelihood function

$$p(\mathbf{y}|\mathbf{X}, \mathbf{p}, \sigma^2) = \prod_{n=1}^N \mathcal{N}(y^{(n)} | \mathbf{p}^T \mathbf{f}(\mathbf{x}^{(n)}), \sigma^2)$$

- Taking log on both sides:

$$\begin{aligned} \ln p(\mathbf{y}|\mathbf{X}, \mathbf{p}, \sigma^2) &= \sum_{n=1}^N \ln \mathcal{N}(y^{(n)} | \mathbf{p}^T \mathbf{f}(\mathbf{x}^{(n)}), \sigma^2) \\ &= -\frac{N}{2} \ln(2\pi) - \frac{N}{2} \ln(\sigma^2) - \frac{1}{2\sigma^2} \sum_{n=1}^N (y^{(n)} - \mathbf{p}^T \mathbf{f}(\mathbf{x}^{(n)}))^2 \end{aligned}$$

- For a linear model with conditional Gaussian noise distribution, maximizing likelihood with respect to weights is equivalent to minimizing the sum-of-squared error.

# Maximum likelihood estimator

- Weights for maximum likelihood – setting the gradient of log likelihood to zero:

$$\nabla \ln p(\mathbf{y}|\mathbf{X}, \mathbf{p}, \sigma^2) = \sum_{n=1}^N (y^{(n)} - \mathbf{p}^T \mathbf{f}(\mathbf{x}^{(n)})) \mathbf{f}(\mathbf{x}^{(n)})^T = 0$$

finally yields

$$\mathbf{p} = (\mathcal{F}^T \mathcal{F})^{-1} \mathcal{F}^T \mathbf{y}$$

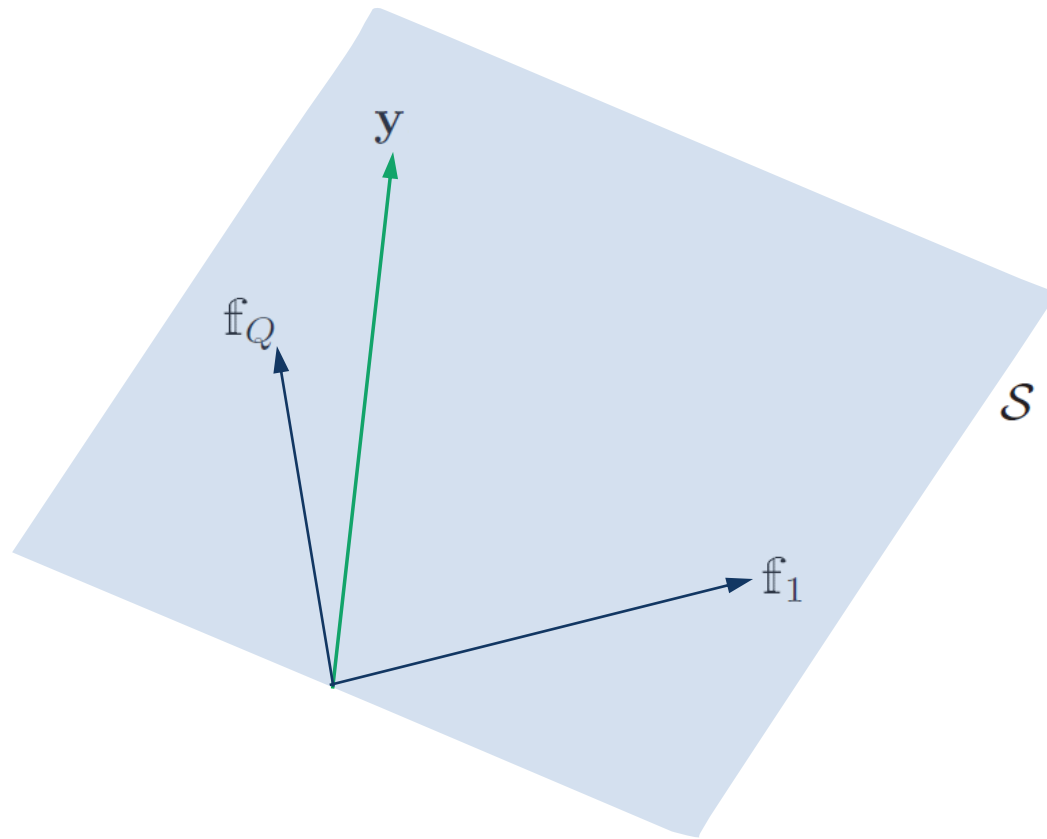
- The noise variance  $\sigma^2$  can also be determined by maximizing (log) likelihood with respect to  $\sigma^2$ , yielding:

$$\sigma^2 = \frac{1}{N} \sum_{n=1}^N (y^{(n)} - \mathbf{p}^T \mathbf{f}(\mathbf{x}^{(n)}))^2$$

- This is the residual variance of the outputs around the predictions

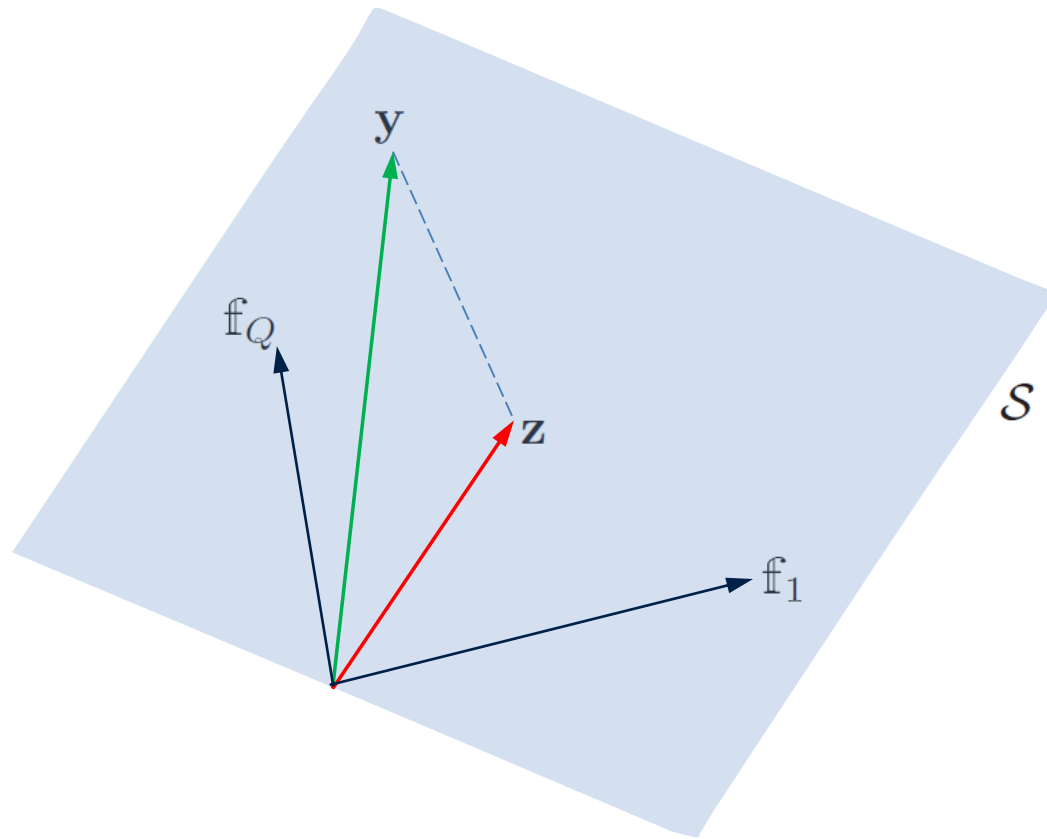
# Geometrical interpretation

- Consider an  $N$ -dimensional space, whose axes indicate the  $N$  output values of the training dataset.
- Let  $\mathbf{f}_j$  be the  $j$ -th column of  $\mathcal{F}$ , i.e.  $\mathbf{f}_j = [f_j(\mathbf{x}^{(1)}), f_j(\mathbf{x}^{(2)}), \dots, f_j(\mathbf{x}^{(N)})]^T$ .
- In the  $N$ -dimensional space consider a subspace  $\mathcal{S}$  spanned by  $Q$  basis vectors  $\{\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_Q\}$ .



# Geometrical interpretation

- Let  $\mathbf{z}$  be an arbitrary linear combination of the basis vectors.
- The vector  $\mathbf{z}$  can lie anywhere in the subspace  $\mathcal{S}$ .
- The least squares algorithm yields solution  $\mathbf{z}$  lying on the subspace  $\mathcal{S}$  which is closest to the vector  $\mathbf{y}$ .



- $R^2$ : Coefficient of determination
- Fraction of the total variation in the outputs that is accounted for by performing regression of the outputs w.r.t. input variables.
- Mathematically  $R^2$  is defined as

$$R^2 = 1 - \frac{S_R}{S_T}$$

- The total variation  $S_T$  is taken as the sum of squared deviations of each  $y$  value from the mean of  $y$ .
  - The mean  $\bar{y}$  is the best guess when the values of the input variables are unknown.
  - The mean  $\bar{y}$  can be computed as
$$\bar{y} = \frac{1}{N} \sum_{n=1}^N y^{(n)}$$
  - The total variation can be thought of as how good the predictions of  $y$  can be without the knowledge of  $\mathbf{x}$ .

- $R^2$ : Coefficient of determination
- Fraction of the total variation in the outputs that is accounted for by performing regression of the outputs w.r.t. input variables.
- Mathematically  $R^2$  is defined as

$$\begin{aligned} R^2 &= 1 - \frac{S_R}{S_T} \\ &= 1 - \frac{\sum_{n=1}^N (z^{(n)} - y^{(n)})^2}{\sum_{n=1}^N (y^{(n)} - \bar{y})^2} \end{aligned}$$

- The sum of square of residuals  $S_R$  is a measure of “unexplained variability” and is defined as

$$S_R = \sum_{n=1}^N (z^{(n)} - y^{(n)})^2$$

- The difference between  $S_T$  and  $S_R$  can be considered as the “explained variability” i.e. the amount of variability in the outputs that is explained by performing regression on the inputs.



# Least mean squares algorithm

- Useful for large datasets, real-time applications.
- Predictions can be made before the whole dataset is seen.
- Updates of  $\mathbf{p}$  are made using the data points in the training set, separately.
- The objective function to be minimized:

$$\begin{aligned} E(\mathbf{p}) &= \sum_{n=1}^N \frac{1}{2} (y^{(n)} - \mathbf{p}^T f(\mathbf{x}^{(n)}))^2 \\ &= \sum_{n=1}^N E_n(\mathbf{p}) \end{aligned}$$

# Stochastic gradient descent

- Initialize the learning rate  $\zeta$  and  $\mathbf{p}$  (say  $\mathbf{p}^{(0)}$ ).
- The weight vector  $\mathbf{p}^{(\tau+1)}$  at the  $\tau$ -th iteration can be updated using the stochastic gradient descent algorithm:

$$\mathbf{p}^{(\tau+1)} = \mathbf{p}^{(\tau)} - \zeta \nabla E_n(\mathbf{p})$$

- On substitution of  $\nabla E_n(\mathbf{p})$  we have

$$\mathbf{p}^{(\tau+1)} = \mathbf{p}^{(\tau)} + \zeta \left( y^{(n)} - (\mathbf{p}^{(\tau)})^T f(\mathbf{x}^{(n)}) \right) f(\mathbf{x}^{(n)})^T$$

# Bayesian linear regression

- The posterior distribution of the weights  $p(\mathbf{p}|\mathbf{y})$  is computed using the Bayes rule

$$p(\mathbf{p}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{p})p(\mathbf{p})}{p(\mathbf{y})}$$

- Assume the prior distribution of the weights  $p(\mathbf{p})$  to be a multivariate Gaussian distribution, i.e.

$$p(\mathbf{p}) = \mathcal{N}(\mathbf{p}|\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$$

where  $\boldsymbol{\mu}_0$  is the mean vector and  $\boldsymbol{\Sigma}_0$  is the covariance matrix.

- If  $\boldsymbol{\Sigma}_0 = \alpha^{-1}\mathbf{I}$ , then the variances of all the weights are equal to  $\alpha^{-1}$ , while the covariances are zero.

- Assume Gaussian likelihood, i.e.

$$p(\mathbf{y}|\mathbf{p}, \mathbf{X}, \beta) = \prod_{n=1}^N \mathcal{N}(y^{(n)}|\mathbf{p}^T \mathbf{f}(\mathbf{x}^{(n)}), \beta^{-1})$$

- Take  $\beta$  (noise precision) to be a fixed constant.

# Bayesian linear regression

- Since the prior  $p(\mathbf{p})$  and the likelihood  $p(\mathbf{y}|\mathbf{p}, \mathbf{X}, \beta)$  are both Gaussian, the posterior distribution  $p(\mathbf{p}|\mathbf{y}, \mathbf{X}, \beta)$  is also Gaussian.
- The posterior distribution is obtained as

$$p(\mathbf{p}|\mathbf{y}, \mathbf{X}, \beta) = \mathcal{N}(\mathbf{p}|\boldsymbol{\mu}_N, \boldsymbol{\Sigma}_N)$$

where

$$\boldsymbol{\mu}_N = \boldsymbol{\Sigma}_N(\boldsymbol{\Sigma}_0^{-1}\boldsymbol{\mu}_0 + \beta\mathcal{F}^T\mathbf{y})$$

is the mean of the posterior distribution and

$$\boldsymbol{\Sigma}_N^{-1} = \boldsymbol{\Sigma}_0^{-1} + \beta\mathcal{F}^T\mathcal{F}$$

is the covariance matrix of the posterior distribution.

- For infinitely broad prior  $\boldsymbol{\Sigma}_0 = \alpha^{-1}\mathbf{I}$  with  $\alpha \rightarrow 0$ , the mean  $\boldsymbol{\mu}_N$  of the posterior distribution reduces to the maximum likelihood solution

$$\boldsymbol{\mu}_N = (\mathcal{F}^T\mathcal{F})^{-1}\mathcal{F}^T\mathbf{y}$$