

# Multivariable Linear Model for Glucose

Attributes

Age, Sex, BMI, BP, S1, ..., S6  
 $x = [x_1, \dots, x_{10}]$

$$y \approx \hat{y} = f(x_1, \dots, x_{10})$$



Target

$y$  = Glucose level

□ Goal: Find a function to predict glucose level from the 10 attributes

□ Linear Model: Assume glucose is a linear function of the predictors:

$$[\text{glucose}] \approx [\text{prediction}] = \beta_0 + \beta_1[\text{Age}] + \dots + \beta_4[\text{BP}] + \beta_5[\text{S1}] + \dots + \beta_{10}[\text{S6}]$$

□ General form:

$$y \approx \hat{y} = \beta_0 + \beta_1 x_1 + \dots + \beta_4 x_4 + \beta_5 x_5 + \dots + \beta_{10} x_{10}$$

↑  
Target              Intercept              10 Features

# Multiple Variable Linear Model

---

- ❑ Vector of **features**:  $x = [x_1, \dots, x_k]$ 
  - $k$  features (also known as predictors, independent variable, attributes, covariates, ...)

- ❑ Single **target variable**  $y$ 
  - What we want to predict

- ❑ Linear model: Make a **prediction**  $\hat{y}$

$$y \approx \hat{y} = \beta_0 + \beta_1 x_1 + \dots + \beta_k x_k$$

- ❑ **Data** for training
  - Samples are  $(x_i, y_i)$ ,  $i=1,2,\dots,n$ .
  - Each sample has a vector of features:  $x_i = [x_{i1}, \dots, x_{ik}]$  and scalar target  $y_i$

- ❑ **Problem**: Learn the best **coefficients**  $\beta = [\beta_0, \beta_1, \dots, \beta_k]$  from the training data

# Example: Heart Rate Increase

□ Linear Model:  $[HR \text{ increase}] \approx \beta_0 + \beta_1[\text{mins exercise}] + \beta_2[\text{exercise intensity}]$

□ Data:

Subject number	HR before	HR after	Mins on treadmill	Speed (min/km)	Days exercise / week
123	60	90	1	5.2	3
456	80	110	2	4.1	1
789	70	130	5	3.5	2
:	:	:	:	:	:



Measuring fitness of athletes

<https://www.mercurynews.com/2017/10/29/4851089/>

# Why Use a Linear Model?

---

□ Many natural phenomena have linear relationship

□ Predictor has small variation

- Suppose  $y = f(x)$
- If variation of  $x$  is small around some value  $x_0$ , then

$$y \approx f(x_0) + f'(x_0)(x - x_0) = \beta_0 + \beta_1 x,$$

$$\beta_0 = f(x_0) - f'(x_0)x_0, \quad \beta_1 = f'(x_0)$$

□ Simple to compute

□ Easy to interpret relation

- Coefficient  $\beta_j$  indicates the importance of feature  $j$  for the target.

□ Advanced: Gaussian random variables:

- If two variables are jointly Gaussian, the optimal predictor of one from the other is linear predictor

# Matrix Review

---

□ Consider

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}, \quad B = \begin{bmatrix} 2 & 0 \\ 3 & 2 \end{bmatrix}, \quad x = \begin{bmatrix} 2 \\ 3 \end{bmatrix},$$

□ Compute (computations on the board):

- Matrix vector multiply:  $Ax$
- Transpose:  $A^T$
- Matrix multiply:  $AB$
- Solution to linear equations: Solve for  $u$ :  $x = Bu$
- Matrix inverse:  $B^{-1}$

# Slopes, Intercept and Inner Products

---

□ Model with **coefficients**  $\beta$ :  $\hat{y} = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k$

□ Sometimes use **weight bias version**:

$$\hat{y} = b + w_1 x_1 + \cdots + w_k x_k$$

- $b = \beta_0$  : Bias or intercept
- $w = \beta_{1:k} = [\beta_1, \dots, \beta_k]$ : Weights or slope vector

□ Can write either with **inner product**:

$$\hat{y} = \beta_0 + \beta_{1:k} \cdot x$$

or

$$\hat{y} = b + w \cdot x$$

□ Inner product:

- $w \cdot x = \sum_{j=1}^k w_j x_j$
- Will use alternate notation:  $w^T x = \langle w, x \rangle$

# Matrix Form of Linear Regression

- Data:  $(x_i, y_i), i = 1, \dots, n$
- Predicted value for  $i$ -th sample:  $\hat{y}_i = \beta_0 + \beta_1 x_{i1} + \dots + \beta_k x_{ik}$

## □ Matrix form

$$\hat{\mathbf{y}} \text{ a } n \text{ predicted values} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_n \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & \cdots & x_{1k} \\ 1 & x_{21} & \cdots & x_{2k} \\ \vdots & \vdots & \cdots & \vdots \\ 1 & x_{n1} & \cdots & x_{nk} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{bmatrix} \quad \beta \text{ with } p = k + 1 \text{ coefficient vector}$$

$\mathbf{A}$  a  $n \times p$  feature matrix

- Matrix equation:

$$\hat{\mathbf{y}} = \mathbf{A} \beta$$