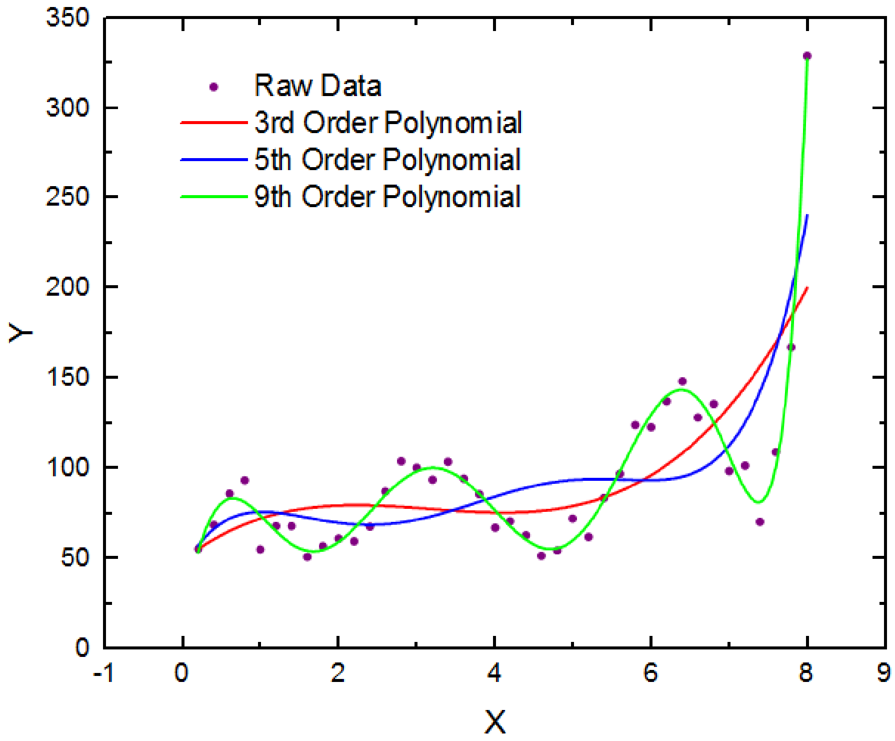
ECE 4252/8803: Fundamentals of Machine Learning (FunML)
Fall 2024

Lecture 7: Linear Regression



OLIVES @GeorgiaTech

Georgia Tech

Reminders

- Reminders
  - HW2 was posted and due this Friday, 13-Sept 8 PM

Overview

In this Lecture..

- Cross-validation
- Precision and Recall
- Confusion Matrix
- Precision/Recall Tradeoff
- The ROC Curve
- Error Analysis

Performance Evaluation

Precision, Recall, and F1 Scores

Precision is a good measure, when the cost of FP is high.

Recall is a good measure, when the cost of FN is high.

F1 is a balance between the two measures and when the classes are uneven.

- **Precision**: how many selected items are relevant? How many of the selected +ve’s are truly +ve?

$$precesion = \frac{TP}{TP + FP} = \frac{3}{5} = 0.6$$

- **Recall**: how many relevant items are selected? How many of the Actual +ve’s has our model labeled as +ve (TP)?

$$recall = \frac{TP}{TP + FN} = \frac{3}{4} = 0.75$$

- **F1 score:** Combines precision and recall:

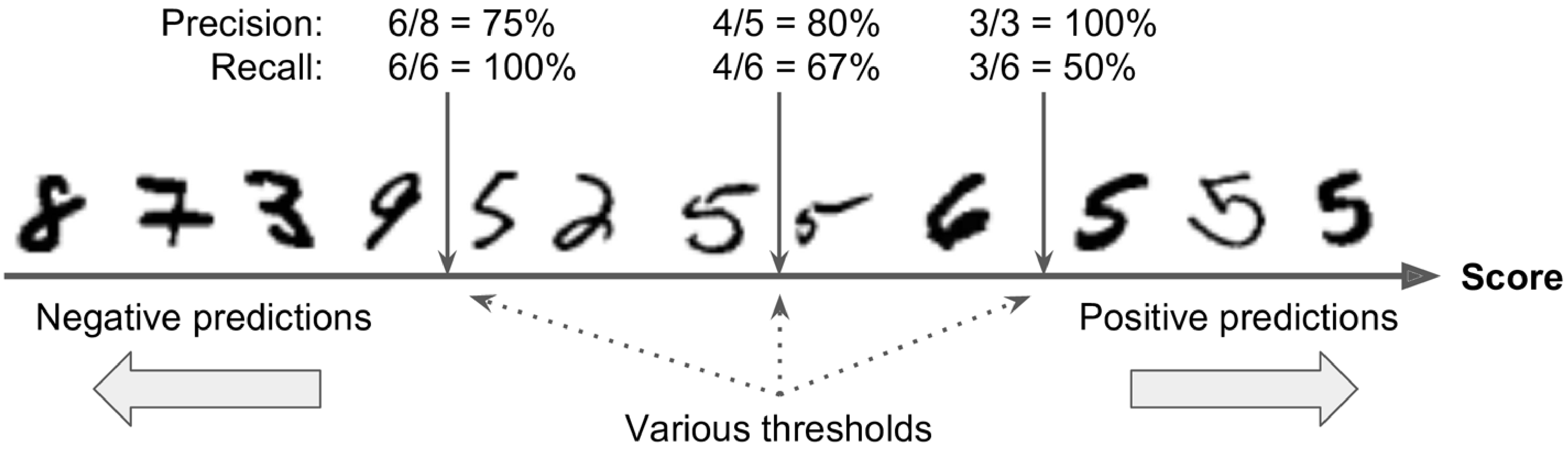
$$F = 2 \times \frac{precision \times recall}{precision + recall} = 2 \times \frac{0.6 \times 0.75}{0.6 + 0.75} = 0.667$$

		Predicted Class	
		P	N
Actual Class	P	True Positives (TP)	False Negatives (FN)
	N	False Positives (FP)	True Negatives (TN)

Q#	Answer Key	Your Answer	Grading	?
1	F	T	✗	FP
2	F	F	✓	TN
3	T	T	✓	TP
4	F	F	✓	TN
5	T	T	✓	TP
6	T	F	✗	FN
7	T	T	✓	TP
8	F	F	✓	TN
9	F	T	✗	FP
10	F	F	✓	TN

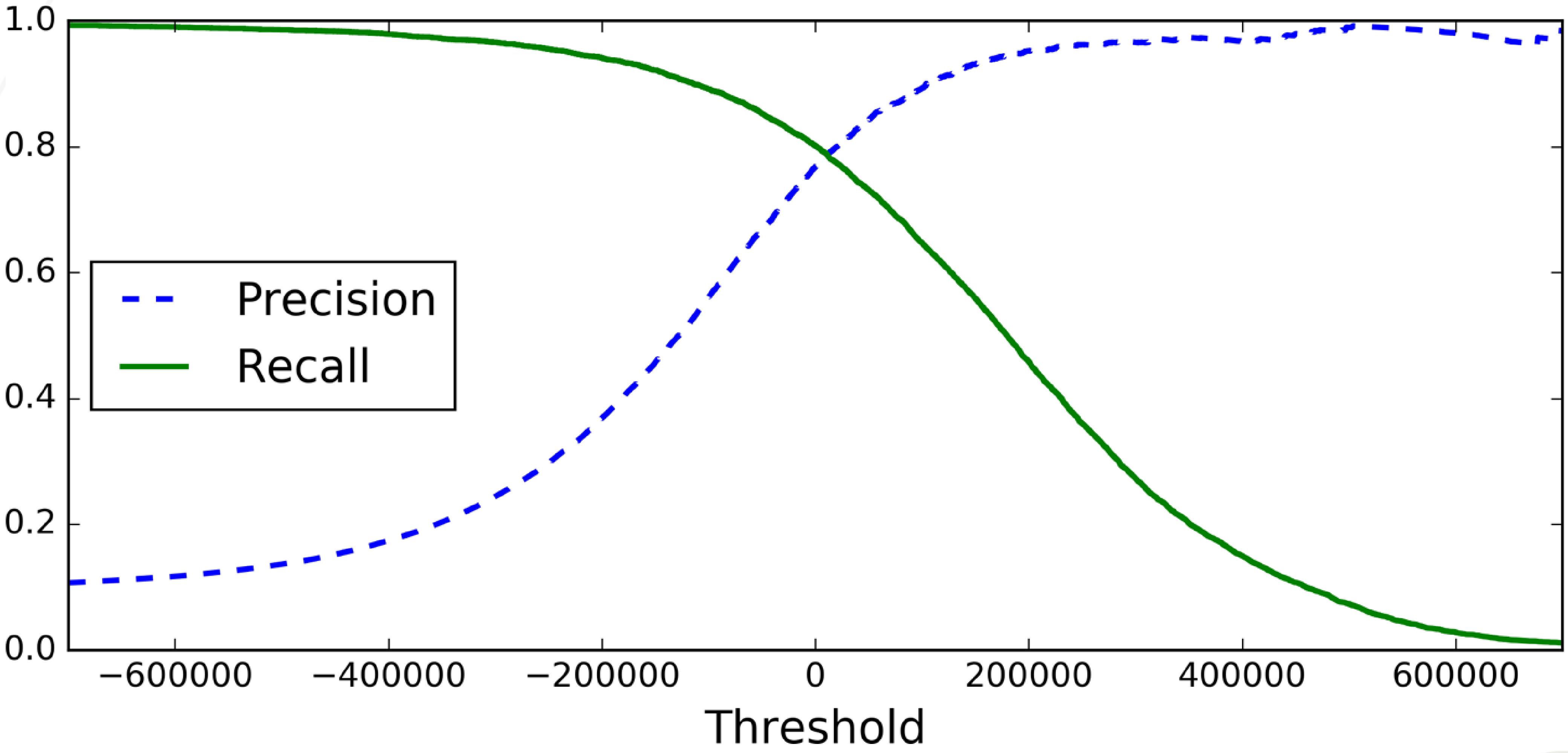
Performance Evaluation in Action

Precision-Recall Tradeoff



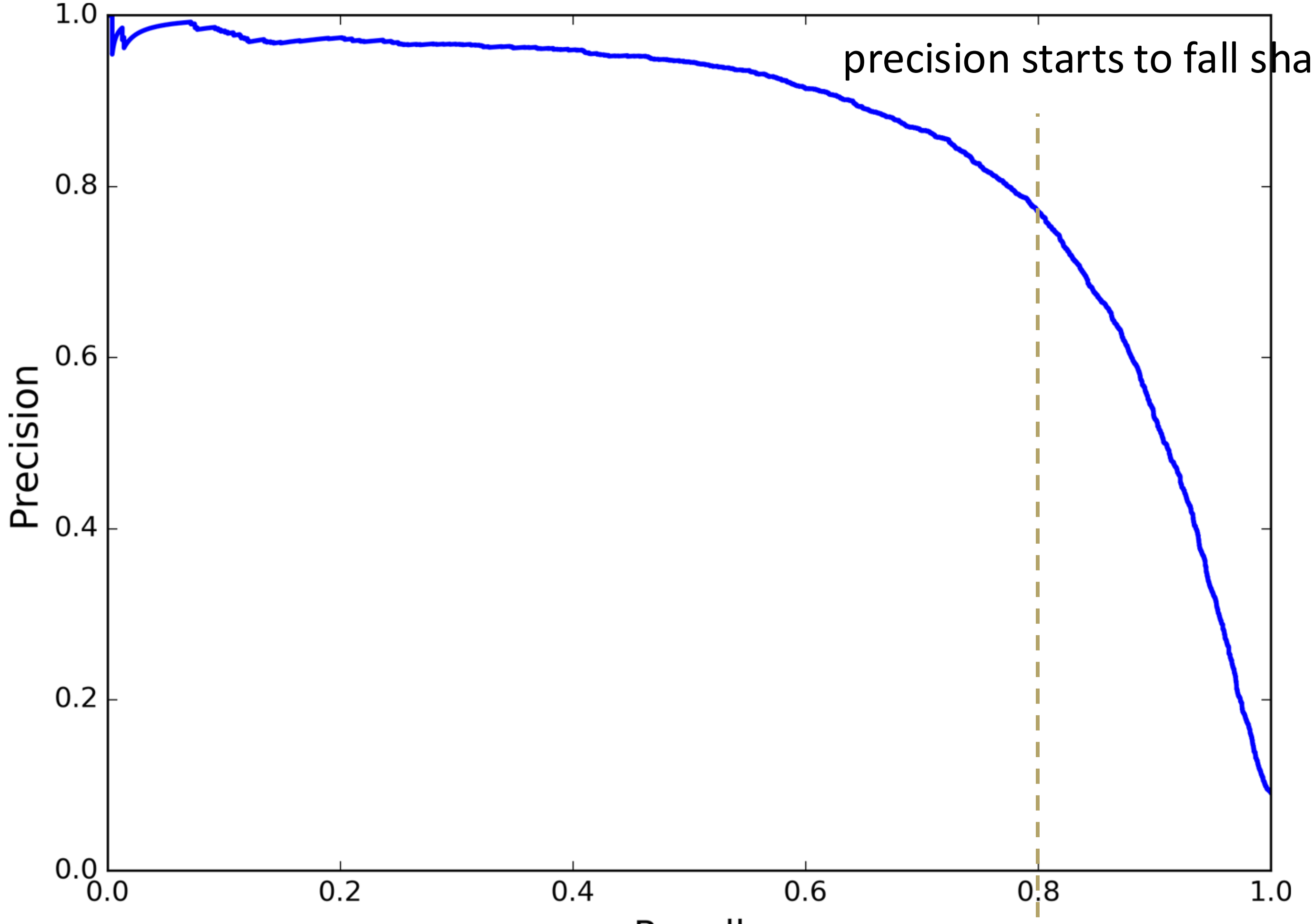
Performance Evaluation in Action

Precision-Recall Tradeoff



Performance Evaluation in Action

Precision-Recall Tradeoff

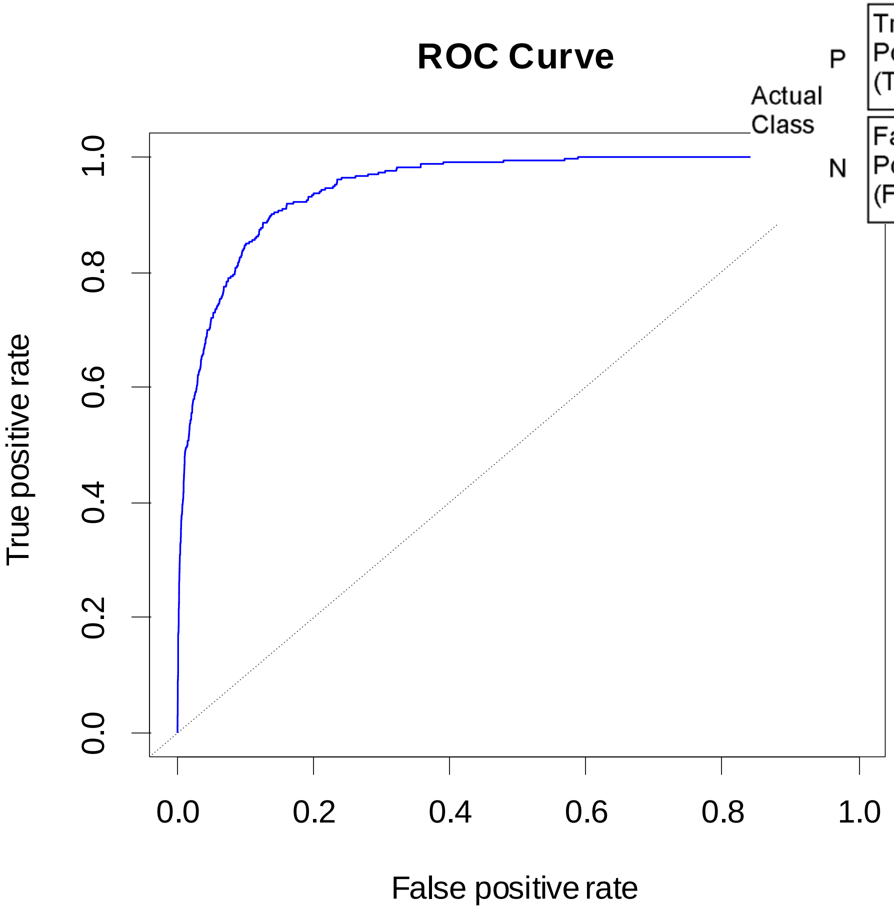


precision starts to fall sharply around 80% recall

Performance Evaluation in Action

ROC Curves

- Receiver Operating Characteristic (ROC), or ROC curve, is a graphical plot that illustrates the performance of classifiers as with varying thresholds.
- The ROC curve is generated by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings.
- TPR (sensitivity/recall): $\mathrm{TPR} = \frac{TP}{TP+FN}$
- FPR (1-specificity): $\mathrm{FPR} = \frac{FP}{FP+TN}$
- Specificity: $\frac{TN}{FP+TN}$
- Sensitivity and specificity are measures of the performance of a binary classification test that are widely used in medicine



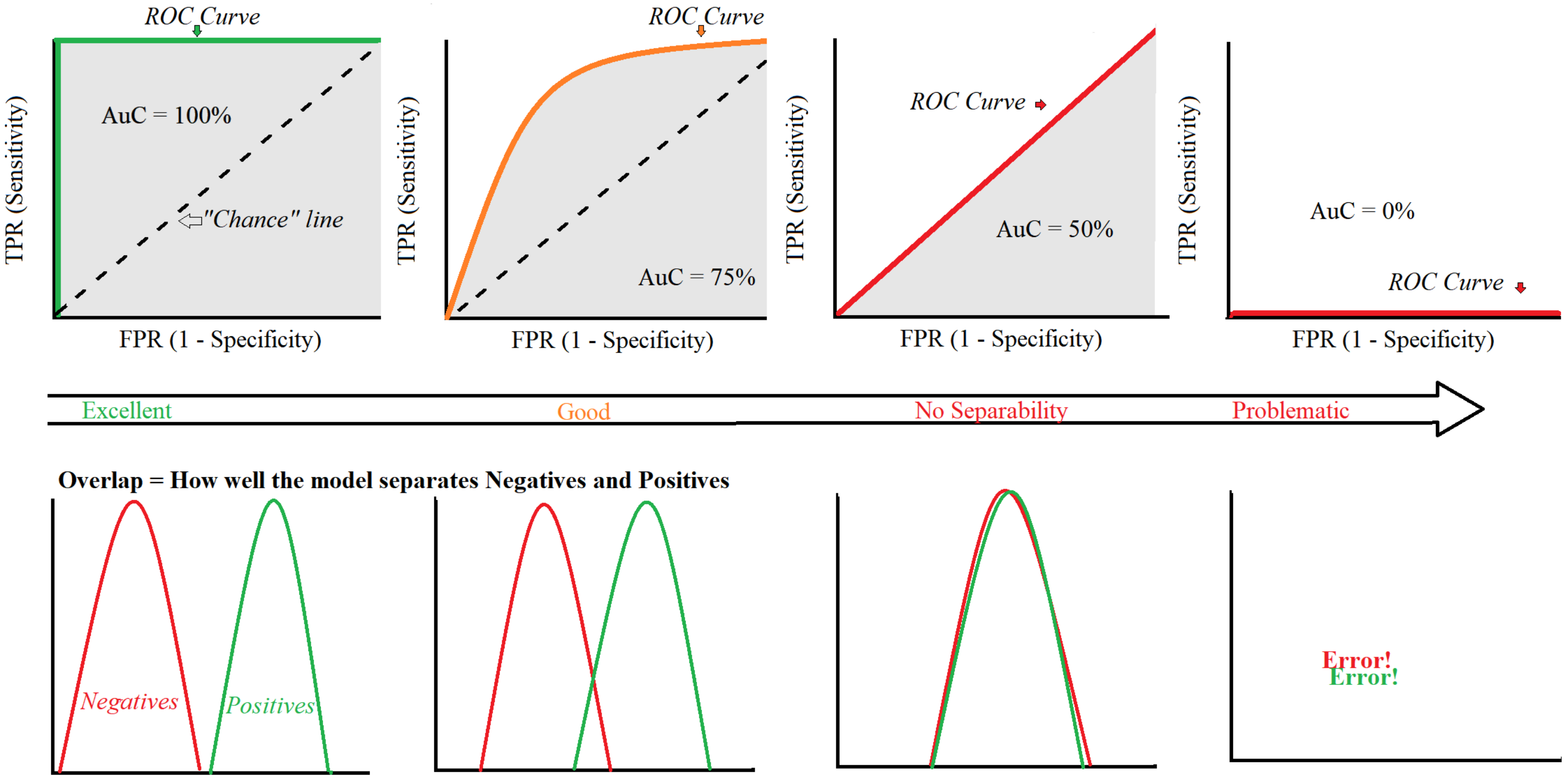
		Predicted Class P	 N
Actual Class	P	True Positives (TP)	False Negatives (FN)
	N	False Positives (FP)	True Negatives (TN)

Sensitivity: the ability of a test to correctly identify patients with a disease.
Specificity: the ability of a test to correctly identify people without the disease.

Performance Evaluation in Action

Area Under the Curve

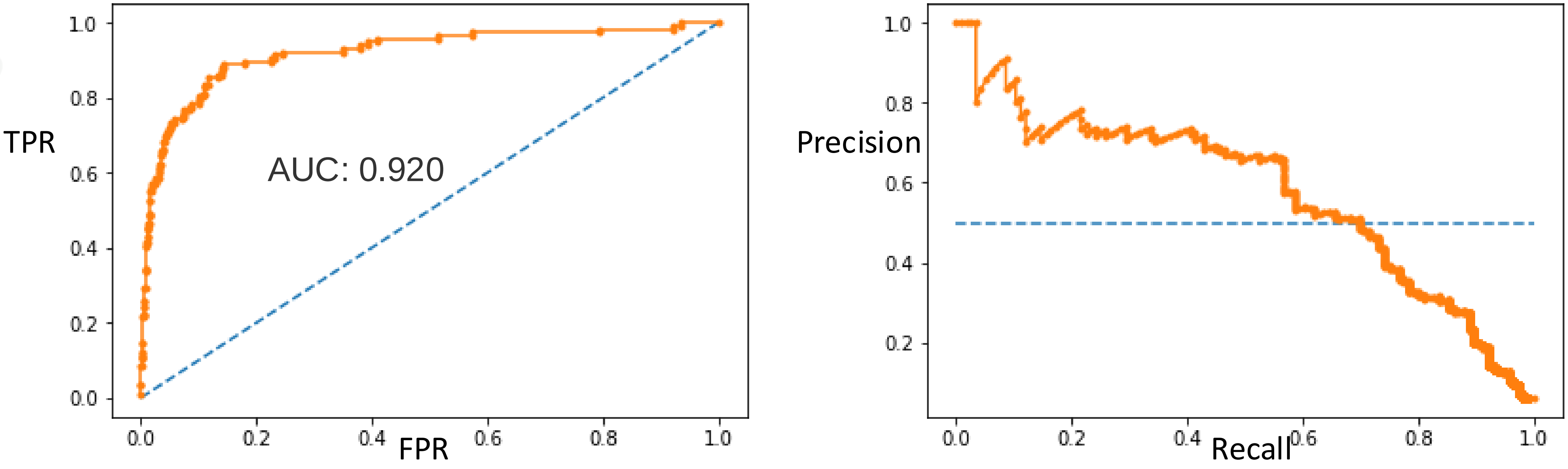
- The area under the curve (AUC) measures the quality of the classifier
- $\mathrm{AUC} \in [0,\ 1]$. The higher the AUC, the better the classifier



Performance Evaluation in Action

Evaluation under Imbalanced Data

- The ROC curve is widely used to evaluate the performance of classifiers. However, it does not measure well for imbalanced data (all classes are not equally represented in test set). And a common alternative is the precision-recall curve.



ROC Curve (left) vs. Precision-Recall Curve (right) with Imbalanced Data (negative class : positive class = 0.95 : 0.05)

Overview

Regression…

Introduction

Linear Regression

Polynomial Regression

Regularization
- Ridge
- Lasso
- Elastic Net

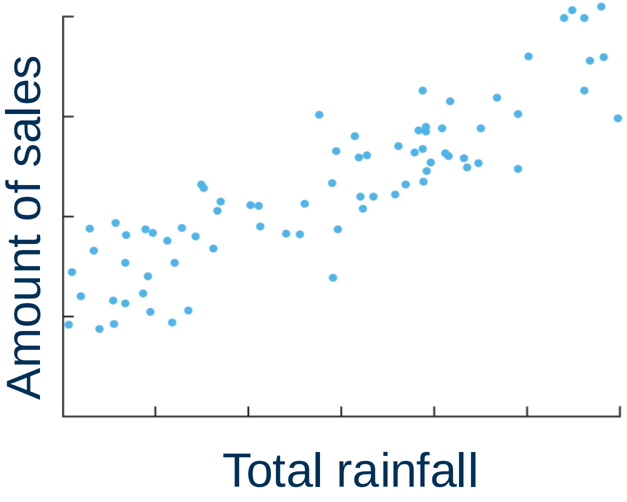
Regularized Linear Models
- Ridge Regression
- Lasso Regression
- Elastic Net Regression

Regression

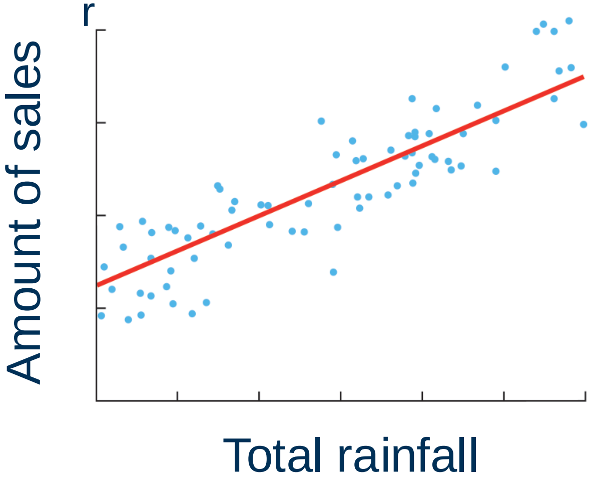
Introduction

- In machine learning, regression is a predictive task for modeling the relationship between input variable $\boldsymbol{x}$ and continuous output variable y by fitting a function $f(\,)$ to observed data. This function is called a predictive regression model.

Is there a relationship between these two variables?



The function $f(x) = 20 + 0.4x$ represented by the red line models a linear r

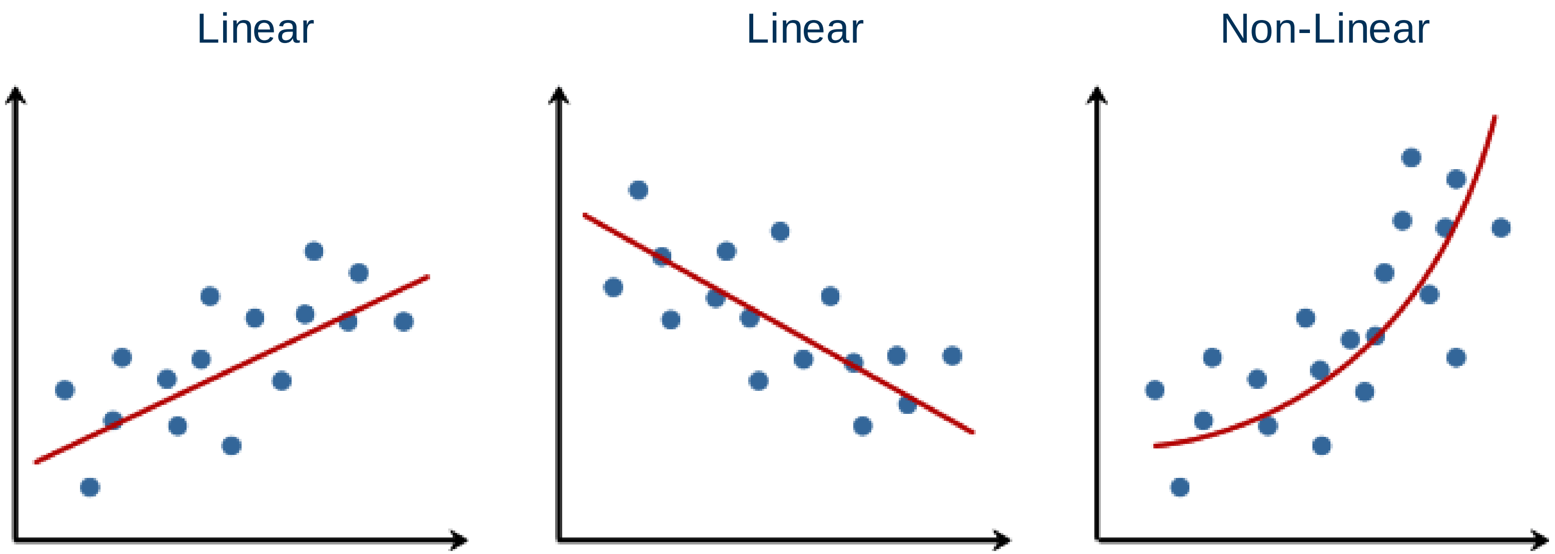


Regression

Introduction

Depending on the relationship between input variables $\boldsymbol{x}$ and the output variable y, two types of regression models can be used:

- **Linear Regression** Models: linear relationship
- **Polynomial Regression** Models: non-linear relationship



Overview

In This Lecture…

Simple Linear Regression Model

General Linear Regression Model

Cost Function

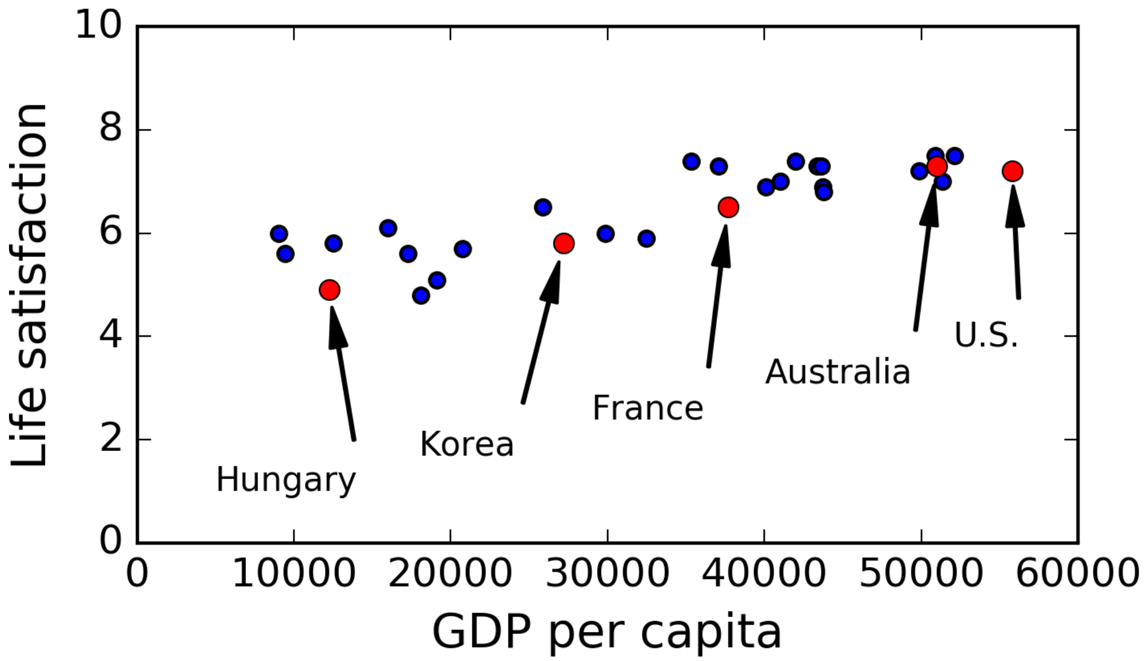
Finding Model Coefficients Using the Normal Equation

Linear Regression

Example

- Let us look at an example of Better Life Index (OECD)
- We observe a trend that Life Satisfaction Index increases *linearly* as GDP per capita increases

Country	GDP per capita (USD)	Life Satisfaction Index
Hungary	12,240	4.9
Korea	27,195	5.8
France	37,675	6.5
Australia	50,962	7.3
United States	55,805	7.2



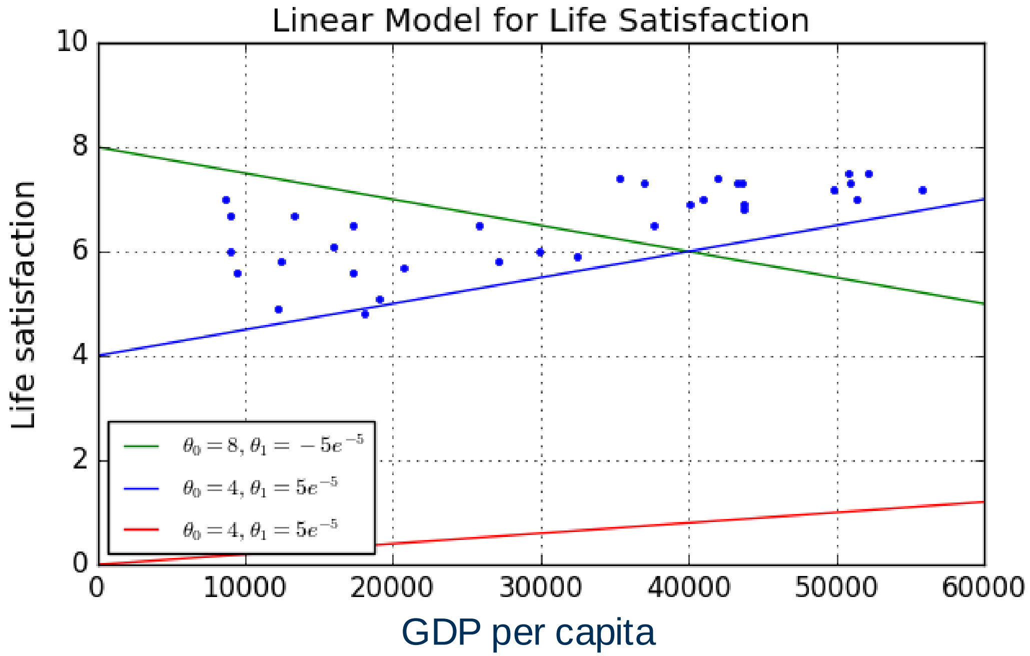
Linear Regression

Example

- In the case where samples are 1-dimensional, a simple linear model fits a *line* to the samples. The line is determined by *intercept* θ_0 and *slope* θ_1 :

$$Life\_Satisfaction \ = \theta_0 + \theta_1 \times GDP\_per\_capita$$

- A few possible model fittings

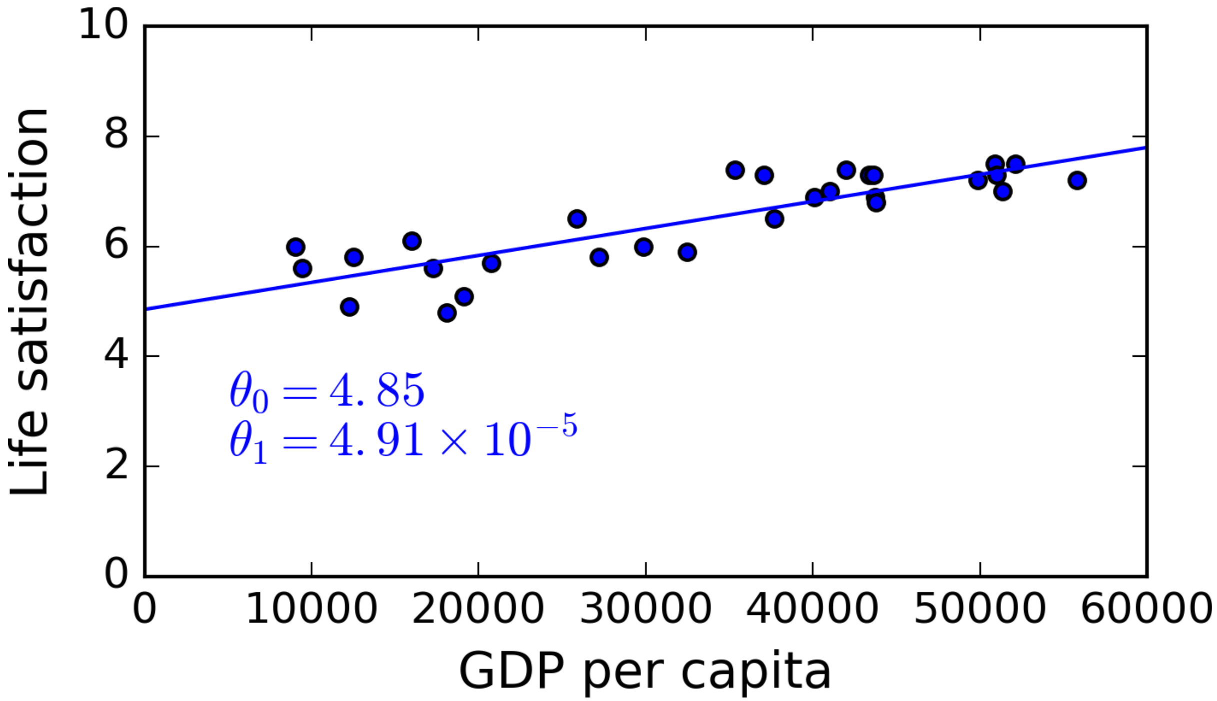


Linear Regression

Example

- The linear regression model that fits the dataset best:

$$Life_{Satisfaction} = 4.85 + (4.91 \times 10^{-5}) \times GDP\_per\_capita$$



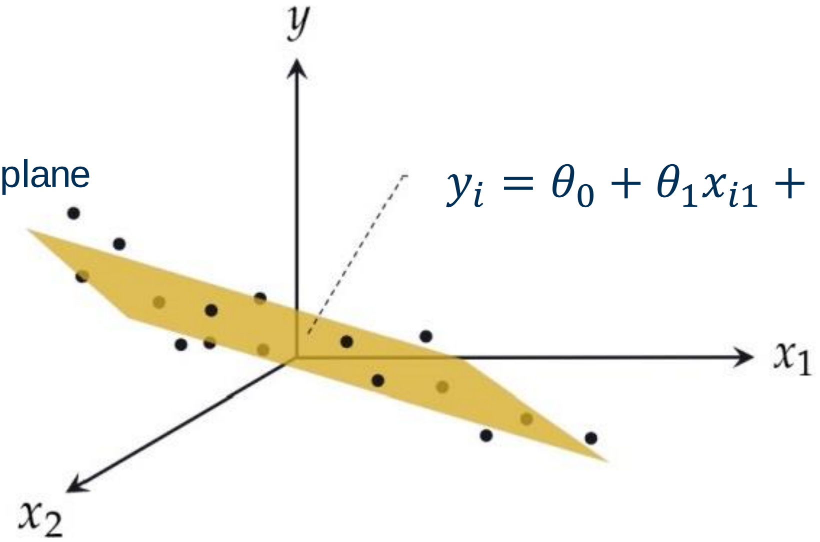
Linear Regression

General Linear Regression Model

- More generally, given a dataset of N samples $(\boldsymbol{x}_1, y_1), \dots (\boldsymbol{x}_N, y_N)$ where input sample $\boldsymbol{x}_i = [x_{i1}, x_{i2}, \dots, x_{iP}]^T \in \mathbb{R}^P$, and the scalar output $y_i \in \mathbb{R}$, a **linear regression** model assumes that the relationship between y_i and its corresponding input $\boldsymbol{x}_i$ is linear.
- The linear model is determined by a **bias** θ_0, and a **weights** vector $\boldsymbol{\theta} = [\theta_1, \theta_2 \dots \theta_P]^T$ to fit a *hyperplane*. The relationship is modeled via the following form:

$$y_i = \theta_0 + \theta_1 x_{i1} + \cdots + \theta_P x_{iP} = \theta_0 + \boldsymbol{\theta}^T \boldsymbol{x}_i, \qquad i = 1, \dots, N$$

Fitting 3-dimensional data with a hyperplane



$y_i = \theta_0 + \theta_1 x_{i1} + \theta_2 x_{i2}$

Linear Regression

General Linear Regression Model

- The linear relationship can be written more compactly, using the notation $\boldsymbol{x}_i = [1, x_{i1}, x_{i2}, \ldots, x_{iP}]^T \in \mathbb{R}^{P+1}$ to denote an input $\boldsymbol{x}_i$ with a 1 placed on top of it. The weights and the bias are placed into a single coefficient vector $\boldsymbol{\theta} = [\theta_0, \theta_1, \theta_2 \ldots \theta_P]^T \in \mathbb{R}^{P+1}$. Then, these N equations are stacked together and written in matrix notation as:

$$\boldsymbol{y} = \boldsymbol{X\theta},$$

Where,

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}, \boldsymbol{x}_i = \begin{bmatrix} 1 \\ x_{i1} \\ \vdots \\ x_{iP} \end{bmatrix}, \boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_N^T \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & \cdots & x_{1P} \\ 1 & x_{21} & \cdots & x_{2P} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N1} & \cdots & x_{NP} \end{bmatrix}, \boldsymbol{\theta} = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_P \end{bmatrix}$$

Linear Regression

General Linear Regression Model

- A linear regression *predictor* aims to estimate this relationship and find *estimated* $\hat{\boldsymbol{y}}$ based on *estimated* coefficients $\hat{\boldsymbol{\theta}}$.
  This is expressed in matrix form as follows

$$\hat{\boldsymbol{y}} = \boldsymbol{X}\hat{\boldsymbol{\theta}} \quad \text{or} \quad \boldsymbol{y} = \boldsymbol{X}\hat{\boldsymbol{\theta}} + \underbrace{\boldsymbol{\varepsilon}}_{\text{Residual error}},$$

where

$$\hat{\boldsymbol{y}} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_N \end{bmatrix}, \boldsymbol{x_i} = \begin{bmatrix} 1 \\ x_{i1} \\ \vdots \\ x_{iP} \end{bmatrix}, \boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_N^T \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & \cdots & x_{1P} \\ 1 & x_{21} & \cdots & x_{2P} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N1} & \cdots & x_{NP} \end{bmatrix}, \hat{\boldsymbol{\theta}} = \begin{bmatrix} \hat{\theta}_0 \\ \hat{\theta}_1 \\ \vdots \\ \hat{\theta}_P \end{bmatrix}, \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_N \end{bmatrix}$$

- Obviously, estimated target vector $\hat{\boldsymbol{y}}$ is a direct result of estimated coefficient vector $\hat{\boldsymbol{\theta}}$. Therefore, the objective in Linear Regression is to find the optimal $\hat{\boldsymbol{\theta}}$

General Linear Regression Model

Least Squares Loss Function

- Recall that a linear regression *predictor* estimates the linear relationship between observed target variable $\boldsymbol{y}$ and input variable $\boldsymbol{X}$ via *estimated* coefficients $\widehat{\boldsymbol{\theta}}$ as following:

$$\boldsymbol{y} = \boldsymbol{X}\widehat{\boldsymbol{\theta}} + \boldsymbol{\varepsilon}$$

Where $\boldsymbol{\varepsilon}$ is the **residual error** between predictions $\widehat{\boldsymbol{y}} = \boldsymbol{X}\widehat{\boldsymbol{\theta}}$ and target $\boldsymbol{y}$

- We often assume that $\boldsymbol{\varepsilon} = \boldsymbol{y} - \widehat{\boldsymbol{y}}$ is i.i.d. (independently and identically distributed) distributed and drawn from a Gaussian (normal) distribution. We denote this by $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mu, \sigma^2)$, where μ is the mean and σ^2 is the variance.
- The PDF of ε_i is given by

$$p(\varepsilon_i) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\varepsilon_i^2}{2\sigma^2}\right)$$

$y_i | \boldsymbol{x}_i; \widehat{\boldsymbol{\theta}} \sim \mathcal{N}(\widehat{\boldsymbol{\theta}}^{\boldsymbol{T}} \boldsymbol{x}_i, \sigma^2)$
y_i is Gaussian distributed given $\boldsymbol{x}_i$

This implies that

$$p(y_i | \boldsymbol{x}_i; \widehat{\boldsymbol{\theta}}) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\left(y_i - \widehat{\boldsymbol{\theta}}^{\boldsymbol{T}} \boldsymbol{x}_i\right)^2}{2\sigma^2}\right)$$

Materials adapted from Murphy, Kevin P. *Machine learning: a probabilistic perspective*. MIT press, 2012.

General Linear Regression Model

Least Squares Loss Function

- For the entire dataset, $p(\boldsymbol{y}|\boldsymbol{X};\widehat{\boldsymbol{\theta}})$ can be viewed as the likelihood function of $\widehat{\boldsymbol{\theta}}$, denoted as $L(\widehat{\boldsymbol{\theta}};\boldsymbol{X},\boldsymbol{y})$
- $\boldsymbol{y}|\boldsymbol{X};\widehat{\boldsymbol{\theta}}$ is i.i.d. because $\boldsymbol{\varepsilon}$ is assumed i.i.d. The likelihood can be written as:
- $L(\widehat{\boldsymbol{\theta}};\boldsymbol{X},\boldsymbol{y}) = \prod_{i=1}^{N} p(y_i|\boldsymbol{x}_i;\widehat{\boldsymbol{\theta}}) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{(y_{\boldsymbol{i}}-\widehat{\boldsymbol{\theta}}^{\boldsymbol{T}}\boldsymbol{x}_i)^2}{2\sigma^2}\right)$

Taking the log probability:

- $\log L(\widehat{\boldsymbol{\theta}};\boldsymbol{X},\boldsymbol{y}) = \sum_{i=1}^{N} \log\frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{(y_{\boldsymbol{i}}-\widehat{\boldsymbol{\theta}}^{\boldsymbol{T}}\boldsymbol{x}_i)^2}{2\sigma^2}\right) = N\log\frac{1}{\sqrt{2\pi\sigma}} - \frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_{\boldsymbol{i}} - \widehat{\boldsymbol{\theta}}^{\boldsymbol{T}}\boldsymbol{x}_i)^2$
- We maximize log probability over all configurations of $\widehat{\boldsymbol{\theta}}$ using Max Likelihood Estimation. Equivalently, we minimize the **negative log-likelihood** $-\log L(\widehat{\boldsymbol{\theta}};\boldsymbol{X},\boldsymbol{y})$, which is equivalent to:

$$\frac{1}{2\sigma^2}\sum_{i=1}^{N}(y_{\boldsymbol{i}} - \widehat{\boldsymbol{\theta}}^{\boldsymbol{T}}\boldsymbol{x}_i)^2$$

Materials adapted from Murphy, Kevin P. *Machine learning: a probabilistic perspective*. MIT press, 2012.

General Linear Regression Model

Least Squares Loss Function

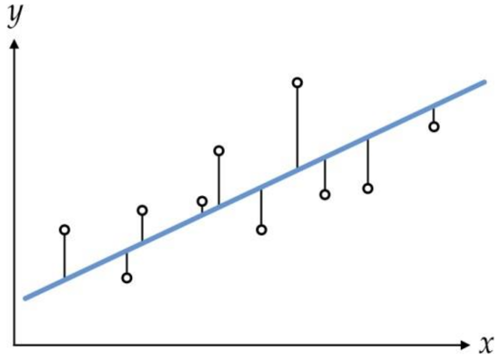
- To find the optimal $\widehat{\boldsymbol{\theta}}$, we minimize the negative log-likelihood $-\log L(\widehat{\boldsymbol{\theta}}; \boldsymbol{X}, \boldsymbol{y})$, which can be viewed as a **loss function** $L(\widehat{\boldsymbol{\theta}})$ that measures the distance between estimated $\widehat{\boldsymbol{y}}$ and target $\boldsymbol{y}$

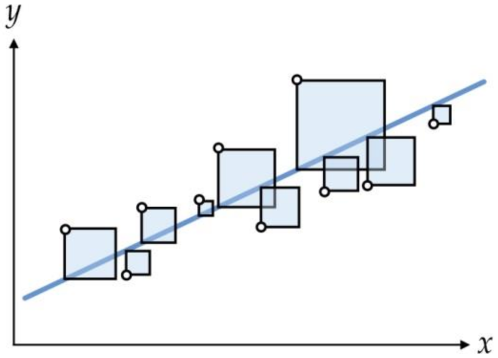
- Such loss function is called Least Squares loss function or Mean Square Error (MSE):

$$L(\widehat{\boldsymbol{\theta}}) = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2 = \frac{1}{N}\sum_{i=1}^{N}(\widehat{\boldsymbol{\theta}}^{\boldsymbol{T}}\boldsymbol{x}_i - y_i)^2$$

- The Least Squares loss function computes the total *squared error* between each model prediction and the associated data sample

Left figure: solid error bars are the difference between samples (dots) and predictions (line)



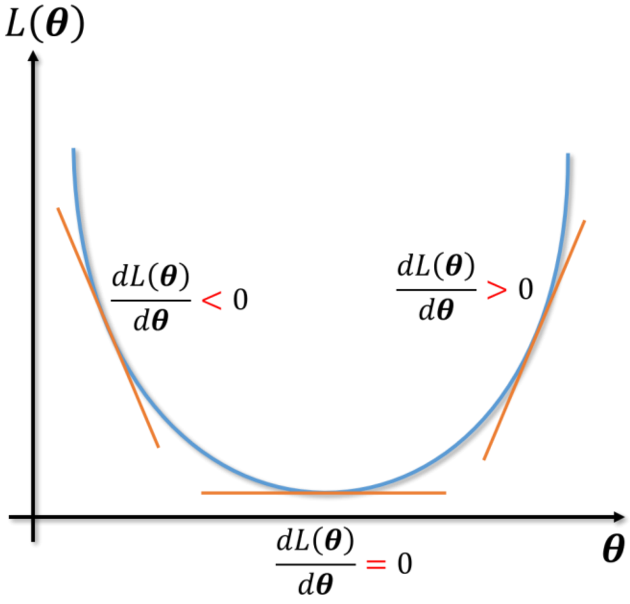


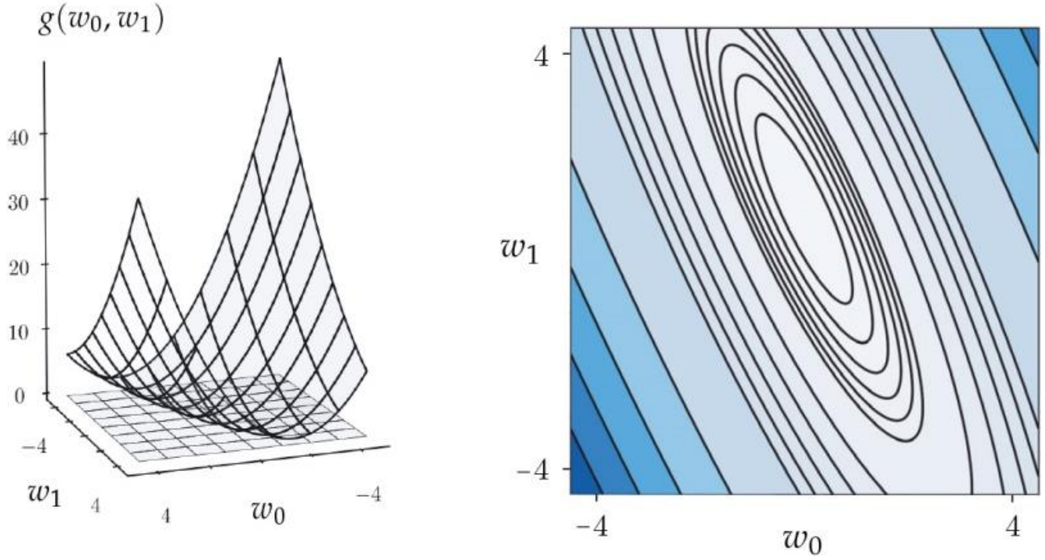
Right figure: The least squares error can be thought as the total area of the blue squares

General Linear Regression Model

Least Squares Loss Function

- Let $L(\widehat{\boldsymbol{\theta}})$ be the loss function that measures the difference between predictions $\widehat{\boldsymbol{y}}$ and desired outputs $\boldsymbol{y}$
- We want to find the **optimum** $\widehat{\boldsymbol{\theta}}^*$ such that:
$$\widehat{\boldsymbol{\theta}}^* = \underset{\widehat{\boldsymbol{\theta}}}{\operatorname{argmin}}\, L(\widehat{\boldsymbol{\theta}})$$
- Considering the change in the loss function with respect to $\widehat{\boldsymbol{\theta}}$, the loss function is minimum when $\frac{dL(\widehat{\boldsymbol{\theta}})}{d\widehat{\boldsymbol{\theta}}} = 0$.
- The Least Squares loss function for linear regression is ***convex quadratic.*** Optimal $\widehat{\boldsymbol{\theta}}$ can be found by solving its **Normal Equation.**


$L(\boldsymbol{\theta})$
$\frac{dL(\boldsymbol{\theta})}{d\boldsymbol{\theta}} < 0$
$\frac{dL(\boldsymbol{\theta})}{d\boldsymbol{\theta}} > 0$
$\frac{dL(\boldsymbol{\theta})}{d\boldsymbol{\theta}} = 0$
$\boldsymbol{\theta}$



General Linear Regression Model

Quick Review

- First order
  - $\frac{\partial(\boldsymbol{x}^T\boldsymbol{a})}{\partial \boldsymbol{x}} = \frac{\partial(\boldsymbol{a}^T\boldsymbol{x})}{\partial \boldsymbol{x}} = \boldsymbol{a}$
- Second order
  - $\frac{\partial(\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x})}{\partial \boldsymbol{x}} = (\boldsymbol{A} + \boldsymbol{A}^T)\boldsymbol{x}$

General Linear Regression Model

Optimal Parameters

- The least squares cost function $\boldsymbol{L}(\hat{\boldsymbol{\theta}}) = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2$ can be written in matrix form as:
  $\boldsymbol{L}(\hat{\boldsymbol{\theta}}) = \frac{1}{N}(\boldsymbol{X}\hat{\boldsymbol{\theta}} - \boldsymbol{y})^T(\boldsymbol{X}\hat{\boldsymbol{\theta}} - \boldsymbol{y})$
- $\boldsymbol{L}(\hat{\boldsymbol{\theta}}) = \frac{1}{N}\left((\boldsymbol{X}\hat{\boldsymbol{\theta}})^T\boldsymbol{X}\hat{\boldsymbol{\theta}} - (\boldsymbol{X}\hat{\boldsymbol{\theta}})^T y - y^T\boldsymbol{X}\hat{\boldsymbol{\theta}} + \boldsymbol{y}^T\boldsymbol{y}\right) = \frac{1}{N}(\hat{\boldsymbol{\theta}}^T\boldsymbol{X}^T\boldsymbol{X}\hat{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}^T\boldsymbol{X}^T\boldsymbol{y} - y^T\boldsymbol{X}\hat{\boldsymbol{\theta}} + \boldsymbol{y}^T\boldsymbol{y})$
- Its derivative is: $\frac{\partial L_{\hat{\boldsymbol{\theta}}}}{\partial\hat{\boldsymbol{\theta}}} = \frac{\partial}{\partial\hat{\boldsymbol{\theta}}}(\hat{\boldsymbol{\theta}}^T\boldsymbol{X}^T\boldsymbol{X}\hat{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}^T\boldsymbol{X}^T\boldsymbol{y} - y^T\boldsymbol{X}\hat{\boldsymbol{\theta}} + \boldsymbol{y}^T\boldsymbol{y})$
- $\frac{\partial}{\partial\hat{\boldsymbol{\theta}}}(\hat{\boldsymbol{\theta}}^T\boldsymbol{X}^T\boldsymbol{X}\hat{\boldsymbol{\theta}}) = (\boldsymbol{X}^T\boldsymbol{X} + \boldsymbol{X}^T\boldsymbol{X})\hat{\boldsymbol{\theta}} = 2\boldsymbol{X}^T\boldsymbol{X}\hat{\boldsymbol{\theta}}$
- $\frac{\partial}{\partial\hat{\boldsymbol{\theta}}}(\hat{\boldsymbol{\theta}}^T\boldsymbol{X}^T\boldsymbol{y}) = \frac{\partial}{\partial\hat{\boldsymbol{\theta}}}(y^T\boldsymbol{X}\hat{\boldsymbol{\theta}}) = \boldsymbol{X}^T\boldsymbol{y}$
- Thus $\boxed{\frac{\partial L_{\hat{\boldsymbol{\theta}}}}{\partial\hat{\boldsymbol{\theta}}} = \frac{1}{N}(2\boldsymbol{X}^T\boldsymbol{X}\hat{\boldsymbol{\theta}} - 2\boldsymbol{X}^T\boldsymbol{y})}$

- When $\frac{\partial L_{\hat{\boldsymbol{\theta}}}}{\partial\hat{\boldsymbol{\theta}}} = 0$:

$$2\boldsymbol{X}^T\boldsymbol{X}\hat{\boldsymbol{\theta}} - 2\boldsymbol{X}^T\boldsymbol{y} = 0$$

$$2\boldsymbol{X}^T\boldsymbol{X}\hat{\boldsymbol{\theta}} = 2\boldsymbol{X}^T\boldsymbol{y}$$

$$(\boldsymbol{X}^T\boldsymbol{X})^{-1}(\boldsymbol{X}^T\boldsymbol{X})\hat{\boldsymbol{\theta}} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

$$\boxed{\hat{\boldsymbol{\theta}} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}}$$

- The above is the Normal Equation which gives a direct solution for the optimal coefficient vector $\hat{\boldsymbol{\theta}}$

General Linear Regression Model

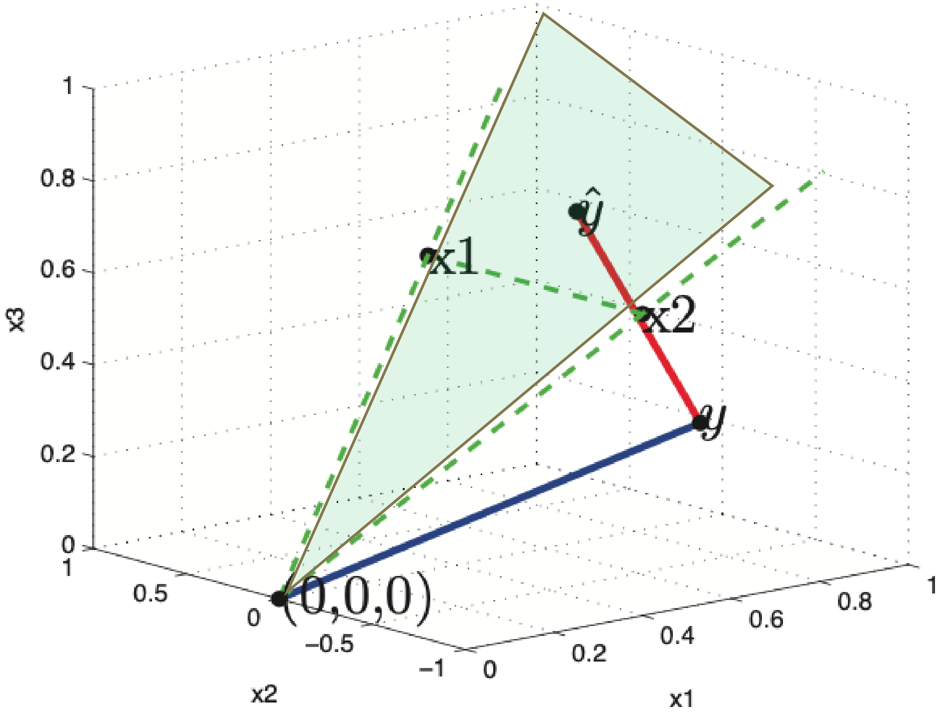
Geometric Interpretation

For a dataset $\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_N^T \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & \cdots & x_{1P} \\ 1 & x_{21} & \cdots & x_{2P} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N1} & \cdots & x_{NP} \end{bmatrix}$, we assume $N > P + 1$, i.e., there are more samples than features.

The columns of $\boldsymbol{X}$ = $[\boldsymbol{1}, \boldsymbol{x}_{:,1} \ldots, \boldsymbol{x}_{:,P}]$ define a (P+1)-dimensional linear subspace. We denote this subspace as $\mathrm{span}\{\boldsymbol{X}\}$ or $\mathrm{span}\{\boldsymbol{1}, \boldsymbol{x}_{:,1} \ldots, \boldsymbol{x}_{:,P}\}$.

We want to estimate $\hat{\boldsymbol{y}} \in \mathbb{R}^N$ that lies in such linear subspace and is as close as possible to $\boldsymbol{y}$:

$$\underset{\hat{\boldsymbol{y}} \in \mathrm{span}\{\boldsymbol{1}, \boldsymbol{x}_{:,1} \ldots, \boldsymbol{x}_{:,P}\}}{\mathrm{argmin}} \|\hat{\boldsymbol{y}} - \boldsymbol{y}\|_2$$



$2D$ linear subspace (green) spanned by $[1,1,1]^T$ and $[1,-1,1]^T$

Materials adapted from Murphy, Kevin P. *Machine learning: a probabilistic perspective*. MIT press, 2012.

General Linear Regression Model

Geometric Interpretation

Since $\hat{\boldsymbol{y}} \in \mathrm{span}\{\mathbf{1}, \boldsymbol{x}_{:,1}\ldots, \boldsymbol{x}_{:,P}\}$, $\hat{\boldsymbol{y}}$ will be the linear combination of $\mathbf{1}, \boldsymbol{x}_{:,1}\ldots, \boldsymbol{x}_{:,P}$ with a coefficient vector $\hat{\boldsymbol{\theta}}$ such that:

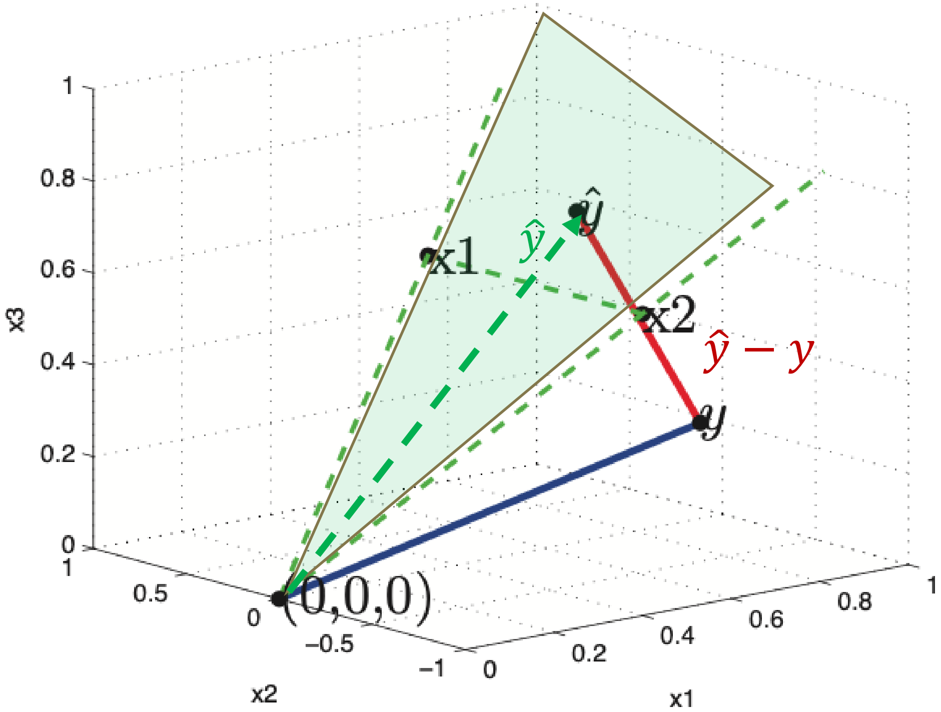
$$\hat{\boldsymbol{y}} = \hat{\theta}_0 + \hat{\theta}_1 \boldsymbol{x}_{:,1} + \cdots + \hat{\theta}_P \boldsymbol{x}_{:,P} = \boldsymbol{X}\hat{\boldsymbol{\theta}}$$

To minimize $\|\hat{\boldsymbol{y}} - \boldsymbol{y}\|_2$, the desired residual vector will be **orthogonal** to $\mathrm{span}\{\mathbf{1}, \boldsymbol{x}_{:,1}\ldots, \boldsymbol{x}_{:,P}\}$, i.e. , $\boldsymbol{x}_{:,j}{}^T\,(\hat{\boldsymbol{y}} - \boldsymbol{y}) = 0, \forall j = 1, \ldots, P$. Its matrix form is $\boldsymbol{X}^T\left(\boldsymbol{X}\hat{\boldsymbol{\theta}} - \boldsymbol{y}\right) = 0$.

Hence, $\hat{\boldsymbol{\theta}} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}$, and

$$\hat{\boldsymbol{y}} = \boldsymbol{X}\hat{\boldsymbol{\theta}} = \boxed{\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\,\boldsymbol{X}^T}\boldsymbol{y}$$

Projection matrix



$\hat{\boldsymbol{y}}$ (green arrow) and $\hat{\boldsymbol{y}} - \boldsymbol{y}$ (red arrow)

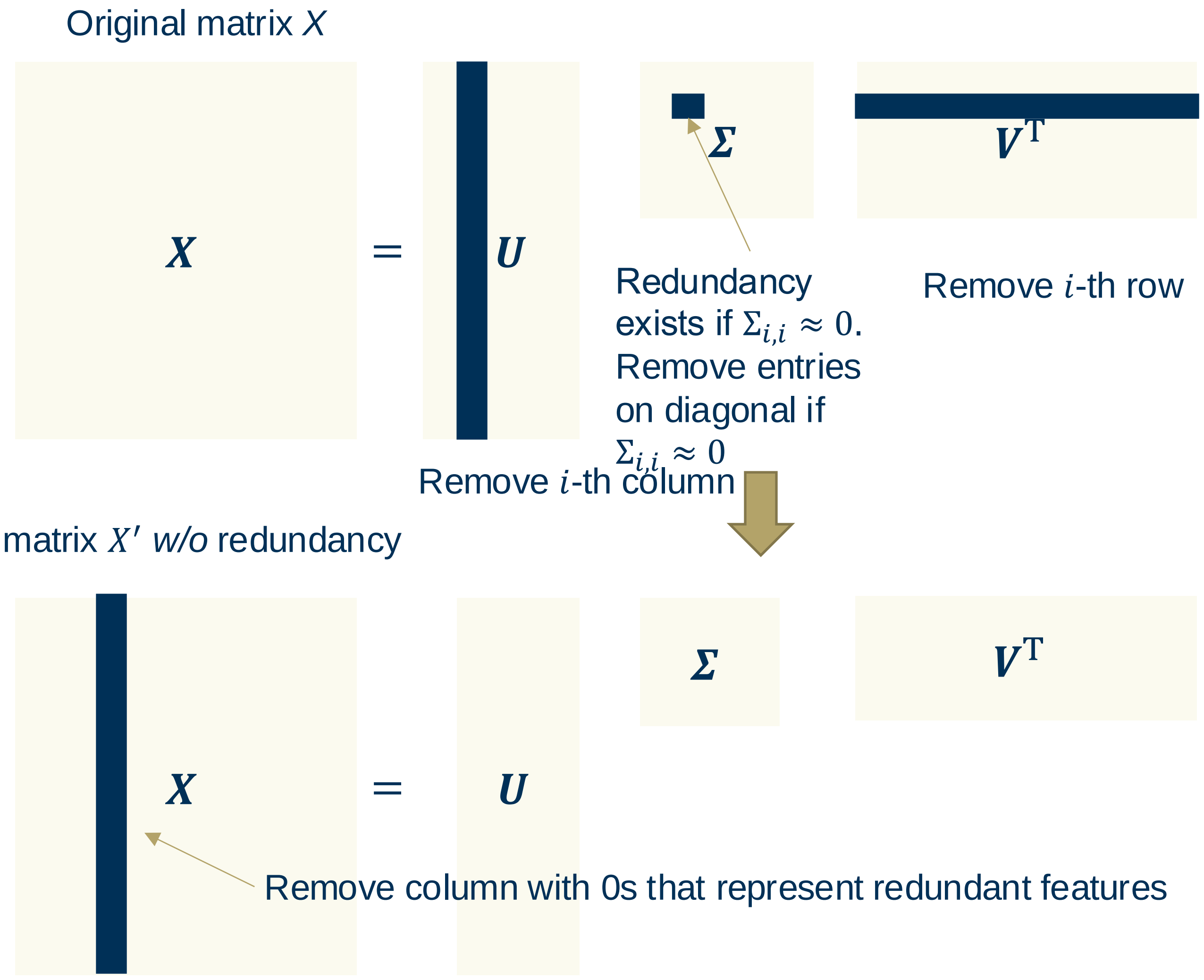
The estimated $\hat{\boldsymbol{y}}$ can be viewed as an **orthogonal projection** of $\boldsymbol{y}$ onto $\mathrm{span}\{\mathbf{1}, \boldsymbol{x}_{:,1}\ldots, \boldsymbol{x}_{:,P}\}$.

Materials adapted from Murphy, Kevin P. *Machine learning: a probabilistic perspective*. MIT press, 2012.

Singular Value Decomposition

Redundant Features

- Recall that we use normal equation to find the optimal $\boldsymbol{\theta}$: $\boldsymbol{\theta} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}$. For a data matrix $X_{N\times P}$, its columns $\{\boldsymbol{x}_{:,0}, \boldsymbol{x}_{:,1} \ldots, \boldsymbol{x}_{:,P}\}$ are linearly independent when none of $\boldsymbol{x}_{:,j}$ can be represented by a weighted sum of the other $\boldsymbol{x}_{:,k}, k \neq j$. In this case, $\boldsymbol{X}^T\boldsymbol{X}$ is invertible thus normal equation works fine.
- However, if some columns are **not linearly independent**, it implies **redundancy or correlation** in our **features** that some of them do not provide independent information for regression. Moreover, using normal equation will cause issues because $\boldsymbol{X}^T\boldsymbol{X}$ is not invertible.
- Singular Value Decomposition (**SVD**) helps us to identify and remove redundant features



Original matrix X

Redundancy exists if $\Sigma_{i,i} \approx 0$. Remove entries on diagonal if $\Sigma_{i,i} \approx 0$

Remove i-th row

Remove i-th column

matrix X' *w/o* redundancy

Remove column with 0s that represent redundant features

Multi-output Regression

In practice, we might encounter data $(\boldsymbol{x}_1, \boldsymbol{y}_1), \dots (\boldsymbol{x}_N, \boldsymbol{y}_N)$ where $\boldsymbol{x}_i = \left[x_{i1}, x_{i2}, \dots, x_{ip}\right]^T \in \mathbb{R}^P$, and the **vector** output $\boldsymbol{y}_i = [y_{i1}, y_{i2}, \dots, y_{iK}]^T \in \mathbb{R}^K$**.**

If we assume a *linear* relationship between $\boldsymbol{x}_i$ and only the *k*-th dimension of y_{ik}**,** it is the regular linear regression we modeled so far: $y_{ik} = \boldsymbol{x}_i \boldsymbol{\theta}$

If we further assume a linear relationship between $\boldsymbol{x}_i$ and **all** K dimensions of the output $\boldsymbol{y}_i \in \mathbb{R}^K$**,** it is called **multi-output regression**.

The relationship is modeled as the following form:

$$\boldsymbol{Y} = \boldsymbol{X\theta}$$

where

$$\boldsymbol{Y} = \begin{bmatrix} \boldsymbol{y}_1^T \\ \boldsymbol{y}_2^T \\ \vdots \\ \boldsymbol{y}_N^T \end{bmatrix} = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1K} \\ y_{21} & y_{22} & \cdots & y_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ y_{N1} & y_{N2} & \cdots & y_{NK} \end{bmatrix}, \boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_N^T \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & \cdots & x_{1P} \\ 1 & x_{21} & \cdots & x_{2P} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N1} & \cdots & x_{NP} \end{bmatrix}, \boldsymbol{\theta} = \begin{bmatrix} \theta_{10} & \theta_{20} & \cdots & \theta_{K0} \\ \theta_{11} & \theta_{21} & \cdots & \theta_{K1} \\ \vdots & \vdots & \ddots & \vdots \\ \theta_{1P} & \theta_{2P} & \cdots & \theta_{KP} \end{bmatrix}$$

Materials adapted from Murphy, Kevin P. *Machine learning: a probabilistic perspective*. MIT press, 2012.

Multi-output Regression

Applications

- Forest properties prediction: predicting forest vegetation height and canopy cover from features derived from satellite imagery data.
- Soil quality prediction: estimating the abundances of Acari and Collembolans, and Shannon-Wiener biodiversity, from agricultural measures.
- Monthly online product sales prediction based on product and advertising features
- Estimating energy consumptions such as heating load and cooling load of residential buildings
- etc.

Appendix A: Notations

- x_i: a single feature
- $\boldsymbol{x}_i$: feature vector (a data sample)
- $\boldsymbol{x}_{:,i}$: feature vector of all data samples
- $\boldsymbol{X}$: matrix of feature vectors (dataset)
- N: number of data samples
- m: degree of polynomial
- P: number of features in a feature vector
- θ_i: a single model coefficient (parameter)
- $\boldsymbol{\theta}$: coefficient vector
- ε: error margin
- α: learning rate
- γ: bias factor
- Bold letter/symbol: vector
- Bold capital letters/symbol: matrix