normal small char -> number
bold small char -> vector
bold capital char -> matrix

1 Linear Algebra

Matrix Operations

- Matrix Multiplication:

$$(AB)_{ij} = \sum_k A_{ik} B_{kj}$$

- Transpose:

$$(A^\top)_{ij} = A_{ji}$$

- Inverse (invertible A):

$$AA^{-1} = A^{-1}A = I$$

- Determinant:

$$\det(A)$$

		Predicted Class		
		Positive	Negative	
Actual Class	Positive	True Positive (TP)	False Negative (FN) Type II Error	Sensitivity $\frac{TP}{(TP+FN)}$
	Negative	False Positive (FP) Type I Error	True Negative (TN)	Specificity $\frac{TN}{(TN+FP)}$
		Precision $\frac{TP}{(TP+FP)}$	Negative Predictive Value $\frac{TN}{(TN+FN)}$	Accuracy $\frac{TP+TN}{(TP+TN+FP+FN)}$

Eigenvalues and Eigenvectors

- Eigenvalue Equation:

عدد ثابت
scale

$$A\mathbf{v} = \lambda\mathbf{v}$$

- Characteristic Equation:

$$\det(A - \lambda I) = 0$$

- Orthogonality (symmetric A):

$$\mathbf{v}_i^\top \mathbf{v}_j = 0 \quad \text{if } i \neq j$$

Matrix Calculus

- Derivative of Linear Form:

$$\frac{\partial}{\partial \mathbf{x}}(\mathbf{a}^\top \mathbf{x}) = \mathbf{a}$$

- Derivative of Quadratic Form (symmetric A):

$$\frac{\partial}{\partial \mathbf{x}}(\mathbf{x}^\top A\mathbf{x}) = 2A\mathbf{x}$$

- Derivative of Determinant:

$$\frac{\partial}{\partial A}\det(A) = \det(A)(A^{-1})^\top$$

- Derivative of Log-Determinant:

$$\frac{\partial}{\partial A}\ln\det(A) = (A^{-1})^\top$$

- Derivative of Matrix Inverse:

$$\frac{\partial A^{-1}}{\partial A} = -A^{-1} \otimes A^{-1}$$

2 Probability and Statistics

Expected Value and Variance

- Expected Value:

$$\mathbb{E}[X] = \int x f_X(x)\, dx \quad \text{or} \quad \mathbb{E}[X] = \sum_i x_i P(X = x_i)$$

- Variance:

$$\mathrm{Var}(X) = \mathbb{E}[(X - \mathbb{E}[X])^2] = \mathbb{E}[X^2] - (\mathbb{E}[X])^2$$

ده → سوا

$$\mathrm{Var}(aX + bY) = a^2\mathrm{Var}(X) + b^2\mathrm{Var}(Y) + 2ab \cdot \mathrm{Cov}(X, Y)$$

- Properties ($Y = aX + b$):

$$\mathbb{E}[Y] = a\mathbb{E}[X] + b$$

$$\mathrm{Var}(Y) = a^2\mathrm{Var}(X)$$

$$\mathrm{Cov}\left(f_i(X), f_j(X)\right) = \rho . \sqrt{Var(X) . Var(Y)}$$

هم بيساوي

Covariance

- Covariance:

$$\mathrm{Cov}(X, Y) = \mathbb{E}[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])] = \mathbb{E}[XY] - \mathbb{E}[X]\mathbb{E}[Y]$$

- Covariance Matrix: ماتريس

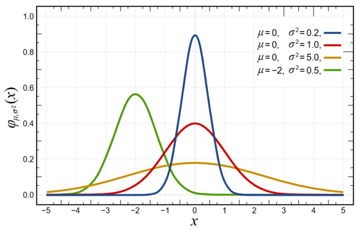
$$\Sigma = \mathbb{E}[(\mathbf{X} - \mathbb{E}[\mathbf{X}])(\mathbf{X} - \mathbb{E}[\mathbf{X}])^\top]$$

- Properties:

$$\mathrm{Cov}(aX + b, Y) = a\mathrm{Cov}(X, Y)$$

$$\mathrm{Cov}(X, Y) = \mathrm{Cov}(Y, X)$$

Gaussian Distribution



- Univariate Gaussian PDF:

$$f_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

- Multivariate Gaussian PDF:

بُعد

$$f_X(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}\underbrace{|\Sigma|}_{\det(\Sigma)}^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mu)^\top \Sigma^{-1}(\mathbf{x} - \mu)\right)$$

Bayes' Theorem

- Formula:

$$P(A|B) = \frac{\overbrace{P(B|A)P(A)}^{P(A,\, B)}}{P(B)}$$

$$P(C_k|x) = \frac{P(x|C_k)P(C_k)}{\sum_{j=1}^{K} P(x|C_j)P(C_j)}$$

3 Calculus

Derivatives

- Exponential Function:

$$\frac{d}{dx}e^{ax} = ae^{ax}$$

$u' e^u$

- Logarithmic Function:

$$\frac{d}{dx}\ln x = \frac{1}{x}$$

$\frac{u'}{u}$

- Sigmoid Function:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

$$\frac{d}{dx}\sigma(x) = \sigma(x)(1 - \sigma(x))$$

Chain Rule

- Composite Functions:

$$\frac{d}{dx}f(g(x)) = f'(g(x)) \cdot g'(x)$$

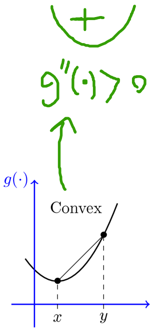
$u'.f'(u)$

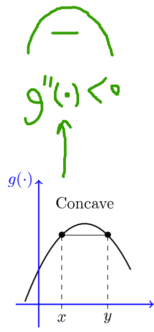
Convex and Concave Functions

محدب مقعر

- Convex Function $\phi : \mathbb{R} \to \mathbb{R}$:

$$\phi(\lambda x + (1-\lambda)y) \leq \lambda\phi(x) + (1-\lambda)\phi(y), \quad \forall x, y \in \mathbb{R}, \quad \forall \lambda \in [0,1]$$





- Concave Function $\phi : \mathbb{R} \to \mathbb{R}$:

$$\phi(\lambda x + (1-\lambda)y) \geq \lambda\phi(x) + (1-\lambda)\phi(y), \quad \forall x, y \in \mathbb{R}, \quad \forall \lambda \in [0,1]$$

4 Optimization

Gradient Descent (Batch Gradient Descent)

- Update Rule:

$$\theta^{(t+1)} = \theta^{(t)} - \alpha \nabla J(\theta^{(t)})$$

where α is the learning rate, $\nabla J(\theta)$ is the gradient. $\left[\frac{\partial J}{\partial \theta_0}, \frac{\partial J}{\partial \theta_1}, \dots, \frac{\partial J}{\partial \theta_n}\right]$ بردار مشتق های جزئی

Stochastic Gradient Descent

- Update Rule:

$$\theta^{(t+1)} = \theta^{(t)} - \alpha \nabla J_i(\theta^{(t)})$$

where $J_i(\theta)$ is the cost for the i-th sample.

5 Machine Learning

Linear Regression

- Model:

نویز

$$y = X\beta + \varepsilon$$

- OLS Estimator:

ordinary least squares

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

- Cost Function:

MSE

$$J(\beta) = \frac{1}{2n} \|y - X\beta\|_2^2$$ میانگین مربع اختلافات پیش بینی با مقادیر واقعی

- Gradient:

$$\nabla J(\beta) = -\frac{1}{n} X^\top (y - X\beta)$$ $u^n \to n u' u^{n-1}$

Logistic Regression

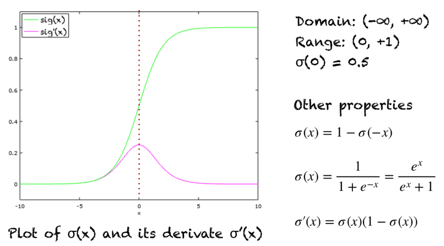
- Model:

$$P(y = 1|x) = \sigma(\beta^\top x)$$

- Sigmoid Function:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

$$\frac{d}{dz}\sigma(z) = \sigma(z)(1 - \sigma(z))$$



Plot of σ(x) and its derivate σ'(x)

- Cost Function:

$$J(\beta) = -\frac{1}{n} \sum_{i=1}^{n} \underbrace{\left[y_i \overset{\log_2}{\ln} \sigma(\beta^\top x_i) + (1 - y_i) \ln(1 - \sigma(\beta^\top x_i))\right]}_{\text{binary cross entropy}}$$

- Gradient:

$$\underset{\partial_\beta}{\nabla J(\beta)} = -\frac{1}{n} X^\top (y - \underset{\hat{y}}{h})$$

where $h = \sigma(X\beta)$.

In summary, L1 (Lasso) and L2 (Ridge) regularization are techniques used to prevent overfitting and improve the generalization of machine learning models by adding a penalty term to the loss function. L1 regularization promotes sparsity in the weights, while L2 regularization promotes smoothness.

Regularization

- L_2 Regularization:

RIDGE

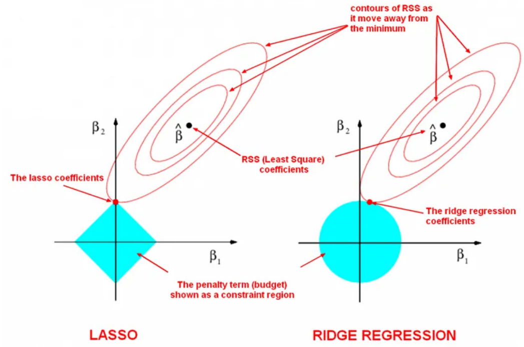
$$J(\beta) = J_0(\beta) + \frac{\lambda}{2}\|\beta\|_2^2$$

$$\nabla J(\beta) = \nabla J_0(\beta) + \lambda\beta$$

- L_1 Regularization:

LASSO

$$J(\beta) = J_0(\beta) + \lambda\|\beta\|_1$$



Lasso can force certain features' coefficients to be zero, thus performing feature selection alongside regularization, while Ridge does not.

Principal Component Analysis (PCA)

- Data Centering:

$$X_c = X - \mu$$

where $\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$.
- Covariance Matrix:

$$\Sigma = \frac{1}{n}X_c^\top X_c$$

Properties:

$$\Sigma = \Sigma^\top, \quad \lambda_i \geq 0, \quad \mathbf{v}^\top\Sigma\mathbf{v} \geq 0$$

$$\mathrm{tr}(\Sigma) = \sum_{i=1}^{p}\lambda_i$$

جمع قطرهای

trace

- Eigenvalue Decomposition:

$$\Sigma\mathbf{v}_i = \lambda_i\mathbf{v}_i$$

where $\overbrace{\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p}^{\text{sort}}$.
- Data Projection:

$$Z = X_c V_k$$

$$PC_1 = V_1^\top X$$

where $V_k = \overbrace{[\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k]}^{\text{first k principal component}}$.
- Explained Variance Ratio:

$$\mathrm{EVR}_i = \frac{\lambda_i}{\sum_{j=1}^{p}\lambda_j}$$

- Maximizing Variance:

$$\max_{\mathbf{w}} \quad \mathbf{w}^\top\Sigma\mathbf{w}$$

subject to:

$$\mathbf{w}^\top\mathbf{w} = 1$$

Leading to:

$$\Sigma\mathbf{w} = \lambda\mathbf{w}$$

- Reconstruction Error:

اونایی که دور ریختم

$$E = \sum_{i=k+1}^{p}\lambda_i$$

- Singular Value Decomposition:

$$X_c = U\Sigma V^\top$$

- Combine the objective function and the constraint using the Lagrange function:

$$\mathcal{L}(x,\lambda) = f(x) + \lambda g(x)$$

- Solve the system of equations:

$$\nabla\mathcal{L}(x,\lambda) = 0$$

- Where:
  - $\mathcal{L}$ is lagrangian
  - λ is lagrange multiplier
  - $g(x)$ is equality constraint
  - $f(x)$ is function

Relation:

$$\lambda_i = \frac{\sigma_i^2}{n}$$

- Matrix Derivatives:

$$\frac{\partial}{\partial \mathbf{w}}(\mathbf{w}^\top \Sigma \mathbf{w}) = 2\Sigma \mathbf{w}$$

$$\frac{\partial}{\partial \Sigma}(\mathbf{w}^\top \Sigma \mathbf{w}) = \mathbf{w}\mathbf{w}^\top$$

K-Means Clustering

- Objective Function:

$$J = \sum_{i=1}^{K} \sum_{x_j \in C_i} \|x_j - \mu_i\|_2^2$$

- Centroid Update:

$$\mu_i = \frac{1}{|C_i|} \sum_{x_j \in C_i} x_j$$

AdaBoost Algorithm

- Initialization:

$$w_i^{(1)} = \frac{1}{n}$$

- Iteration ($t = 1, 2, \ldots, T$):

$$\varepsilon_t = \sum_{i=1}^{n} w_i^{(t)} I(h_t(x_i) \neq y_i)$$

$$\alpha_t = \frac{1}{2} \ln\left(\frac{1 - \varepsilon_t}{\varepsilon_t}\right)$$

$$w_i^{(t+1)} = w_i^{(t)} \exp(-\alpha_t y_i h_t(x_i))$$

Normalize $w_i^{(t+1)}$.

Bias-Variance Decomposition

- Mean Squared Error:

$$\text{MSE}(x) = \text{Bias}(x)^2 + \text{Var}(x) + \overbrace{\sigma^2}^{\text{نویز}}$$

- Bias:

$$\text{Bias}(x) = \mathbb{E}_{\mathcal{D}}[\hat{f}(x)] - f(x)$$ = میانگین مدل ها منهای تابع اصلی

- Variance:

$$\text{Var}(x) = \mathbb{E}_{\mathcal{D}}\left[(\hat{f}(x) - \mathbb{E}_{\mathcal{D}}[\hat{f}(x)])^2\right]$$ = واریانس مدل ها

6 Other Mathematical Concepts

Jensen's Inequality

- For Convex Function ϕ:

$$\phi(\overbrace{\mathbb{E}[X]}^{\mu_x}) \leq \mathbb{E}[\phi(X)]$$

- For Concave Function ϕ:

$$\phi(\underbrace{\mathbb{E}[X]}_{\mu_x}) \geq \mathbb{E}[\phi(X)]$$