Mathematics for Machine Learning: Essential Equations (V4)

1. Basic Linear Algebra

- **Scalar Multiplication:**

$$\mathbf{c} \cdot \mathbf{v} = \begin{bmatrix} cv_1 \\ cv_2 \\ \vdots \\ cv_n \end{bmatrix}$$

- **Matrix-Vector Multiplication:**

$$\mathbf{A} \cdot \mathbf{v} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}$$

- **Norm of a Vector:**

$$||\mathbf{v}|| = \sqrt{v_1^2 + v_2^2 + \cdots + v_n^2}$$

- **Dot Product:**

$$\mathbf{u} \cdot \mathbf{v} = \sum_{i=1}^{n} u_i v_i$$

- **Cross Product (3D Vectors):**

$$\mathbf{u} \times \mathbf{v} = \begin{bmatrix} u_2 v_3 - u_3 v_2 \\ u_3 v_1 - u_1 v_3 \\ u_1 v_2 - u_2 v_1 \end{bmatrix}$$

- **Outer Product:**

$$\mathbf{u} \otimes \mathbf{v} = \begin{bmatrix} u_1 v_1 & u_1 v_2 & \cdots & u_1 v_n \\ u_2 v_1 & u_2 v_2 & \cdots & u_2 v_n \\ \vdots & \vdots & \ddots & \vdots \\ u_m v_1 & u_m v_2 & \cdots & u_m v_n \end{bmatrix}$$

- **Matrix Addition:**

$$\mathbf{A} + \mathbf{B} = \begin{bmatrix} a_{11} + b_{11} & \cdots & a_{1n} + b_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} + b_{m1} & \cdots & a_{mn} + b_{mn} \end{bmatrix}$$

- **Matrix Multiplication:**
$$(\mathbf{A} \cdot \mathbf{B})_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$$
- **Transpose of a Matrix:**
$$(\mathbf{A}^T)_{ij} = a_{ji}$$
- **Inverse of a Matrix (for square A):**
$$\mathbf{A}^{-1} \cdot \mathbf{A} = \mathbf{I}, \quad \text{where } \mathbf{I} \text{ is the identity matrix.}$$

2. Basic Probability and Statistics

- **Conditional Probability:**
$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$
- **Law of Total Probability:**
$$P(A) = \sum_{i} P(A \cap B_i) = \sum_{i} P(A|B_i)P(B_i)$$
- **Bayes' Theorem:**
$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$
- **Expectation:**
$$\mathbb{E}[X] = \sum_{i} x_i P(x_i) \quad \text{(discrete)} \quad \text{or} \quad \int xp(x)dx \quad \text{(continuous)}$$
- **Variance:**
$$\text{Var}(X) = \mathbb{E}[(X - \mathbb{E}[X])^2]$$
- **Standard Deviation:**
$$\sigma = \sqrt{\text{Var}(X)}$$
- **Covariance:**
$$\text{Cov}(X, Y) = \mathbb{E}[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])]$$
- **Correlation Coefficient:**
$$\rho_{X,Y} = \frac{\text{Cov}(X, Y)}{\sigma_X \sigma_Y}$$
- **Probability Mass Function (PMF):**
$$P(X = x) = p(x), \quad \sum_{x} p(x) = 1$$
- **Probability Density Function (PDF):**
$$\int_{-\infty}^{\infty} p(x)dx = 1 \quad \text{for continuous random variables.}$$

3. Basic Calculus

- **Derivative of a Function:**

$$\frac{d}{dx}[f(x)] = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

- **Partial Derivatives:**

$$\frac{\partial f}{\partial x} = \lim_{h \to 0} \frac{f(x+h, y) - f(x, y)}{h}$$

- **Gradient:**

$$\nabla f(\mathbf{x}) = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix}$$

- **Chain Rule:**

$$\frac{dy}{dx} = \frac{dy}{du} \cdot \frac{du}{dx}$$

- **Second Derivative (Hessian Matrix):**

$$\mathbf{H}(f) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

- **Taylor Series Expansion:**

$$f(x) \approx f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \cdots$$

- **Gradient Descent Update Rule:**

$$\mathbf{w} \leftarrow \mathbf{w} - \eta \nabla J(\mathbf{w})$$

- **Optimization Objective:**

$$\min_{\mathbf{x}} f(\mathbf{x})$$

- **Logarithmic Derivative:**

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

- **Exponential Derivative:**

$$\frac{d}{dx}[e^x] = e^x$$

4. Basic Optimization

- **Gradient Descent:**

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \nabla J(\mathbf{w}_t)$$

- **Learning Rate Decay:**

$$\eta_t = \frac{\eta_0}{1 + \lambda t}$$

- **Stochastic Gradient Descent (SGD):**

$$\mathbf{w} \leftarrow \mathbf{w} - \eta \nabla J(\mathbf{w}; x_i, y_i)$$

- **Momentum-based Optimization:**

$$v_t = \beta v_{t-1} + (1 - \beta) \nabla J(\mathbf{w}), \quad \mathbf{w} \leftarrow \mathbf{w} - \eta v_t$$

- **Nesterov Accelerated Gradient (NAG):**

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \nabla J(\mathbf{w}_t + \beta(\mathbf{w}_t - \mathbf{w}_{t-1}))$$

- **RMSProp:**

$$\mathbf{w} \leftarrow \mathbf{w} - \frac{\eta}{\sqrt{\nabla^2 J(\mathbf{w}) + \epsilon}} \nabla J(\mathbf{w})$$

- **Adam Optimization:**

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) \nabla J(\mathbf{w}), \quad v_t = \beta_2 v_{t-1} + (1 - \beta_2)(\nabla J(\mathbf{w}))^2$$

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}, \quad \hat{v}_t = \frac{v_t}{1 - \beta_2^t}$$

$$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t - \frac{\eta \hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}$$

- **Regularized Optimization Objective:**

$$J(\mathbf{w}) = \text{Loss}(\mathbf{w}) + \lambda ||\mathbf{w}||^2$$

- **Projection Gradient Descent:**

$$\mathbf{w}_{t+1} = \Pi_C(\mathbf{w}_t - \eta \nabla J(\mathbf{w}_t)) \quad \text{where } \Pi_C \text{ projects onto set } C$$

- **Newton's Method:**

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \mathbf{H}^{-1} \nabla J(\mathbf{w}_t) \quad \text{where } \mathbf{H} \text{ is the Hessian matrix.}$$

5. Basic Regression Equations

- **Linear Regression Hypothesis:**

$$\hat{y} = \mathbf{X} \cdot \mathbf{w} + b$$

- **Mean Absolute Error (MAE):**

$$\text{MAE} = \frac{1}{m} \sum_{i=1}^{m} |y_i - \hat{y}_i|$$

- **Mean Squared Error (MSE):**

$$\text{MSE} = \frac{1}{m} \sum_{i=1}^{m} (y_i - \hat{y}_i)^2$$

- **Ridge Regression Objective:**

$$J(\mathbf{w}) = \frac{1}{m} \sum_{i=1}^{m} (\hat{y}_i - y_i)^2 + \lambda \sum_{j=1}^{n} w_j^2$$

- **Lasso Regression Objective:**

$$J(\mathbf{w}) = \frac{1}{m} \sum_{i=1}^{m} (\hat{y}_i - y_i)^2 + \lambda \sum_{j=1}^{n} |w_j|$$

- **Logistic Regression Hypothesis:**

$$\hat{y} = \sigma(\mathbf{X} \cdot \mathbf{w} + b), \quad \sigma(z) = \frac{1}{1 + e^{-z}}$$

- **Binary Cross-Entropy Loss:**

$$J(\mathbf{w}) = -\frac{1}{m} \sum_{i=1}^{m} \left[y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \right]$$

- **Coefficient of Determination (R-squared):**

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

- **Adjusted R-squared:**

$$\bar{R}^2 = 1 - \frac{(1 - R^2)(n - 1)}{n - p - 1}$$

- **Gradient of the MSE Loss:**

$$\nabla J(\mathbf{w}) = \frac{1}{m} \mathbf{X}^T (\mathbf{X}\mathbf{w} - \mathbf{y})$$

6. Basic Neural Network Concepts

- **Perceptron Update Rule:**

$$\mathbf{w} \leftarrow \mathbf{w} + \eta(y - \hat{y})\mathbf{x}$$

- **Sigmoid Activation Function:**

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

- **ReLU Activation Function:**

$$f(x) = \max(0, x)$$

- **Softmax Function:**

$$\text{Softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}}$$

- **Loss Function for Multi-Class Classification:**

$$J(\mathbf{w}) = -\frac{1}{m} \sum_{i=1}^{m} \sum_{k=1}^{K} y_{ik} \log(\hat{y}_{ik})$$

- **Forward Propagation (Single Layer):**

$$a = \sigma(\mathbf{w}^T \mathbf{x} + b)$$

- **Backward Propagation (Gradient for Weights):**

$$\frac{\partial J}{\partial \mathbf{w}} = \mathbf{x}(\hat{y} - y)$$

- **Gradient Descent for Neural Networks:**

$$\mathbf{w} \leftarrow \mathbf{w} - \eta \frac{\partial J}{\partial \mathbf{w}}$$

- **Dropout Regularization:**

$$h_i^{(l)} = r_i h_i^{(l)}, \quad r_i \sim \text{Bernoulli}(p)$$

- **Batch Normalization:**

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}, \quad y_i = \gamma \hat{x}_i + \beta$$

7. Basic Clustering Concepts

- **k-Means Objective Function:**

$$J = \sum_{k=1}^{K} \sum_{i \in C_k} ||x_i - \mu_k||^2$$

- **Centroid Update Rule:**

$$\mu_k = \frac{1}{|C_k|} \sum_{x \in C_k} x$$

- **Distance Metric (Euclidean Distance):**

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

- **Silhouette Score:**

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$

- **DBSCAN Core Point Condition:**

$$|N_\epsilon(x)| \geq \text{MinPts} \quad \text{where } N_\epsilon(x) = \{y : d(x, y) \leq \epsilon\}$$

- **Hierarchical Clustering Dendrogram Objective:**

$$\text{Minimize the linkage criterion } L(A, B)$$

- **Gaussian Mixture Model (GMM):**

$$p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x | \mu_k, \Sigma_k)$$

- **Expectation-Maximization (E-step):**

$$\gamma_{ik} = \frac{\pi_k \mathcal{N}(x_i | \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_i | \mu_j, \Sigma_j)}$$

- **Expectation-Maximization (M-step):**

$$\mu_k = \frac{\sum_{i=1}^{N} \gamma_{ik} x_i}{\sum_{i=1}^{N} \gamma_{ik}} \quad \text{and} \quad \Sigma_k = \frac{\sum_{i=1}^{N} \gamma_{ik} (x_i - \mu_k)(x_i - \mu_k)^T}{\sum_{i=1}^{N} \gamma_{ik}}$$

- **Elbow Method for Optimal k:**

$$\text{Choose } k \text{ where } J(k) \text{ has the largest drop.}$$

8. Basic Dimensionality Reduction Concepts

- **Principal Component Analysis (PCA) Objective:**

$$\text{Maximize } ||\mathbf{Xw}||^2 \text{ subject to } ||\mathbf{w}|| = 1$$

- **Covariance Matrix for PCA:**

$$\mathbf{C} = \frac{1}{m}\mathbf{X}^T\mathbf{X}$$

- **Eigen Decomposition for PCA:**

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

- **t-SNE Objective:**

$$C = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

- **Singular Value Decomposition (SVD):**

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$$

- **LDA Objective (Fisher's Criterion):**

$$J(\mathbf{w}) = \frac{\mathbf{w}^T\mathbf{S}_b\mathbf{w}}{\mathbf{w}^T\mathbf{S}_w\mathbf{w}}$$

- **Reconstruction Error for PCA:**

$$\text{Error} = ||\mathbf{X} - \hat{\mathbf{X}}||_F$$

- **Kernel PCA Transformation:**

$$\phi(\mathbf{x}) \rightarrow \text{Principal Components in Feature Space}$$

- **Autoencoder Reconstruction:**

$$\mathbf{X} \approx g(f(\mathbf{X}))$$

- **Explained Variance Ratio:**

$$\text{Ratio} = \frac{\lambda_i}{\sum_j \lambda_j}$$

9. Basic Probability Distributions

- **Bernoulli Distribution:**

$$P(X = x) = p^x(1-p)^{1-x}, \quad x \in \{0, 1\}$$

- **Binomial Distribution:**

$$P(X = k) = \binom{n}{k} p^k (1-p)^{n-k}, \quad k \in \{0, 1, \ldots, n\}$$

- **Poisson Distribution:**

$$P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!}, \quad k \geq 0$$

- **Uniform Distribution:**

$$f(x) = \begin{cases} \frac{1}{b-a}, & a \leq x \leq b \\ 0, & \text{otherwise} \end{cases}$$

- **Normal Distribution:**

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- **Exponential Distribution:**

$$f(x) = \begin{cases} \lambda e^{-\lambda x}, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

- **Beta Distribution:**

$$f(x; \alpha, \beta) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha, \beta)}, \quad x \in [0, 1]$$

- **Gamma Distribution:**

$$f(x; \alpha, \beta) = \frac{\beta^\alpha x^{\alpha-1} e^{-\beta x}}{\Gamma(\alpha)}, \quad x \geq 0$$

- **Multinomial Distribution:**

$$P(X_1 = x_1, \ldots, X_k = x_k) = \frac{n!}{x_1! x_2! \cdots x_k!} p_1^{x_1} p_2^{x_2} \cdots p_k^{x_k}$$

- **Chi-Square Distribution:**

$$f(x; k) = \frac{x^{\frac{k}{2}-1} e^{-\frac{x}{2}}}{2^{\frac{k}{2}} \Gamma(\frac{k}{2})}, \quad x \geq 0$$

10. Basic Reinforcement Learning Concepts

- **Bellman Equation for State-Value Function:**

$$V(s) = \mathbb{E}[R_t + \gamma V(S_{t+1})|S_t = s]$$

- **Bellman Equation for Action-Value Function:**

$$Q(s, a) = \mathbb{E}[R_t + \gamma Q(S_{t+1}, A_{t+1})|S_t = s, A_t = a]$$

- **Policy Improvement:**

$$\pi'(s) = \arg\max_a Q(s, a)$$

- **Temporal Difference Update Rule:**

$$V(S_t) \leftarrow V(S_t) + \alpha[R_{t+1} + \gamma V(S_{t+1}) - V(S_t)]$$

- **Q-Learning Update Rule:**

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha[R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t)]$$

- **SARSA Update Rule:**

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha[R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)]$$

- **Reward Function:**

$$R(s, a) = \mathbb{E}[R_t|S_t = s, A_t = a]$$

- **Value Iteration Update Rule:**

$$V(s) \leftarrow \max_a [R(s, a) + \gamma \sum_{s'} P(s'|s, a)V(s')]$$

- **Actor-Critic Policy Update:**

$$\theta \leftarrow \theta + \alpha \nabla_\theta \log \pi_\theta(a|s)\delta$$

- **Discounted Return:**

$$G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$