Machine Learning 1 - Cheat Sheet

Multivariate Calculus

Index notation

- $[\mathbf{Av}]_i = \sum_p \mathbf{A}_{ip}\mathbf{v}_p$
- $\mathbf{v}^{\mathrm{T}}\mathbf{Ax} = \sum_p \sum_q \mathbf{v}_p\mathbf{A}_{pq}\mathbf{x}_q$
- $\mathbf{v}^{\mathrm{T}}\mathbf{x} = \sum_p \mathbf{v}_p\mathbf{x}_p$

Multivariate derivatives

- $\frac{\partial}{\partial \mathbf{w}}\mathbf{w}^{\mathrm{T}}\mathbf{v} = \mathbf{v}^{\mathrm{T}}$
- $\frac{\partial}{\partial \mathbf{w}}\mathbf{w}^{\mathrm{T}}\mathbf{Aw} = \mathbf{w}^{\mathrm{T}}(\mathbf{A} + \mathbf{A}^{\mathrm{T}})$
- $\frac{\partial}{\partial \mathbf{w}}\mathbf{Aw} = \mathbf{A}$

Useful functions

- Kronecker delta: $\delta_{ik} = \begin{cases} 1 & \text{if } i = k \\ 0 & \text{otherwise} \end{cases}$
- Indicator function: $\mathbf{1}_A(x) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A \end{cases}$

Conventions

- Vectors are columns ($\mathbf{x} \in \mathbb{R}^{n\times 1}$)
- If $f : \mathbb{R}^n \longrightarrow \mathbb{R}^m$, then $\nabla f \in \mathbb{R}^{m\times n}$

Probability & Statistics

Probability

- Sum rule: $P(X) = \sum_Y P(X,Y)$ (disc.)
- Product rule: $P(X,Y) = P(X \mid Y)P(Y)$
- Bayes rule: $P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)}$
- X, Y are independent $\Leftrightarrow P(X,Y) = P(X)P(Y)$

Moments

- $\mathbb{E}[f(X)] = \int_x f(x)p(x)dx$ (cont.)
- $\mathbb{E}[f(X)] = \sum_x f(x)p(x)$ (disc.)
- $\mathrm{Var}[X] = \mathbb{E}[(X - \mathbb{E}[X])^2] = \mathbb{E}[X^2] - \mathbb{E}[X]^2$
- $\mathrm{Cov}[X,Y] = \mathbb{E}\big[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])\big]$ $= \mathbb{E}[XY] - \mathbb{E}[X]\mathbb{E}[Y]$

Distributions

- Univariate Normal: $N(x|\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$
- Multivariate Normal: $N(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^k|\boldsymbol{\Sigma}|}}e^{-\frac{1}{2}(\mathbf{x}-\mu)^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}$
- Uniform: $\frac{1}{b-a}$, $a \le x \le b$

Estimation

- MLE: $\hat{\mathbf{w}}_{\mathrm{ML}} = \arg\max_{\mathbf{w}}\ p(\mathbf{D} \mid \mathbf{w})$
- MAP: $\hat{\mathbf{w}}_{\mathrm{MAP}} = \arg\max_{\mathbf{w}}\ p(\mathbf{w} \mid \mathbf{D})$

Optimization

- Gradient descent: $\mathbf{w}^{t+1} = \mathbf{w}^t - \eta\nabla_{\mathbf{w}} f$

Regression

Linear Regression with Basis Functions

- Model: $t = \mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}) + \boldsymbol{\varepsilon}$, $\boldsymbol{\varepsilon} \sim \mathcal{N}(0,\beta^{-1})$
- Least sq. sol.: $\hat{\mathbf{w}} = \left(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}$
- Reg. least sq. sol.: $\hat{\mathbf{w}} = \left(\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi} + \lambda\mathbf{I}\right)^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}$

where

- Design matrix: $\boldsymbol{\Phi} = (\boldsymbol{\phi}(\mathbf{x}_1), \boldsymbol{\phi}(\mathbf{x}_2), \ldots)^{\mathrm{T}}$

Classification

Naive Bayes assumption

$$p(\mathbf{x}|C_k) = \prod_{d=1}^{D} p(x_d|C_k)$$

- One-hot trick: $p(\mathbf{x}|\boldsymbol{t}) = \prod_{k=1}^{K}(p(\mathbf{x}|t_k = 1))^{t_k}$
  For selecting the correct probability distribution given a one-hot encoded vector $\boldsymbol{t}$.

Logistic Regression

- Sigmoid function: $\sigma(z) = \frac{1}{1+e^{-z}}$
- Softmax function: $\boldsymbol{\varsigma}(\mathbf{z})_i = \frac{\exp z_i}{\sum_{j=1}^{n}\exp z_j}$

Cross-entropy loss

$$E = -\sum_{n=1}^{N}\sum_{k=1}^{K} y_{nk}\log(\hat{y}_{nk})$$

with $\mathbf{y}_n = (y_{n1}, y_{n2}, \ldots, y_{nK})^T$ a one-hot encoding of the true label, and $\hat{\mathbf{y}}_n$ the vector of predicted class probabilities.