PRML functions documentation:

- Kernel matrix(K) formula: (gamma ($X_1 X^T_2$) + coeff)^{degree} where X_1 and X_2 are feature matrix. For degree = 1, this kernel will behave as a linear kernel.
  Default values: gamma = 1, coeff = 0, degree = 1 (Unless otherwise stated explicitly.)
- add_intercept: It will add one's column in feature matrix X whose shape is:
  (no of samples x no of features)
- $y_{predicted} = K(diag(y_{train})\alpha)$ for classification (additional parameters and functions(like round ,sign or sigmoid) will be applied according to algorithm) and $y_{predicted} = K\alpha$ for regression where α is a dual coefficient of shape (no of training samples,) and K is a kernel matrix of shape (no of test samples x no of training samples) while prediction and (no of training samples x no of training samples) while training.
- Log loss : $J = -[y_{true}^T \log(y_h) + (1 - y_{true}^T)\log(1 - y_h)]\,/m$
- Log loss gradient: $\frac{\partial J}{\partial \alpha} = \frac{diag(y_{train})K(y_h - y_{train})}{m}$
- Log loss Hessian: $\frac{\partial^2 J}{\partial \alpha^2} = \frac{diag(y_{train})Kdiag(y_h)(1-diag(y_h))Kdiag(y_{train})}{m}$
  Where $y_h = \sigma(K(diag(y_{train})\alpha))$
- Theoretical Perceptron Bound used in the report:

$$r = \max(\{||x_i|| \ : i = 1, \dots, m\}) = \max(diag(K)) \ \dots\dots (1)$$

$$\rho = \min\left(\frac{diag(y)X\omega}{||\omega||}\right) = \min\left(\frac{diag(y)XX^T(diag(y)\alpha)}{(\alpha^T diag(y))XX^T(diag(y)\alpha)}\right)$$

$$= \min\left(\frac{diag(y)K(diag(y)\alpha)}{(\alpha^T diag(y))K(diag(y)\alpha)}\right) \ \dots\dots (2)$$

$$\delta = \sqrt{\sum_t d_t^2} \ where \ d_t = \max\{0, \rho - y_t(v \cdot \ x_t)\} \ \dots\dots (3)$$

$$v \ can \ be \ any \ unitvector, for \ simplicity \ v = \left[\frac{1}{\sqrt{m}}, \frac{1}{\sqrt{m}}, \dots, \frac{1}{\sqrt{m}}\right]_{mx1}$$

$$Bound = \left(\frac{r + \delta}{\rho}\right)^2 \ \dots\dots (4)$$

  ***Note:** To keep ρ>0 all such x_i are not considered in equation 2 which gives non positive values due to non-separability. And as r is norm, and there exists at least one x in the dataset, which is not a zero vector, so r>0 condition is automatically satisfied. Also, equation 4 gives general bound for non-linearly separable sample, so kernel modifications are implemented in the above equations like replacing K for XX^T or diag(K) for $\{||x_i|| \ : i = 1, \dots, m\}$.
- Equation for kernel ridge regression: $(K + \lambda I)K\alpha = K^T y$
- Other default values: (Unless otherwise stated explicitly.)

  T = 100 (perceptron epochs), b = 0 (bias for perceptron), learning rate = 1 (for perceptron and newton method), learning rate = 0.01 for gradient descent, iterations =1000 (for gradient descent), C = 1.0(for soft margin SVM), C = 1e10(for hard margin SVM), initial dual coefficients = zero vector of shape:(no of samples,) (For perceptron algorithm too.)