

1 Dokumentacja

Dokumentacja projekt SVM quadratic

1.1 class SVMClassifier

class SVMClassifier - klasa klasyfikatora

Zmienne protected klasy SVMClassifier:

double C - wartość C typu double, ustawiona na 1.0

double epsilon - wartosc epsilon ustawiona na 0.0001

char fNameTrain[256] - nazwa pliku z próbkami treningowymi, maksymalna długość 256

char fNameTest[256] - nazwa pliku z próbkami testowymi, maksymalna długość 256

char fNameResults[256] - nazwa pliku z rezultatami, maksymalna długość 256

int N - liczba próbek treningowych początkowa wartość 0

int NTestSamples - liczba próbek testowych początkowa wartość 0

TVectorArray arrayX - macierz próbek treningowych

TFloatArray arrayY - macierz próbek testowych

TFloatArray alpha - wektor z wartosciami alpha, początkowa inicjalizacja zerami

TFloatArray d - macierz pomocnicza w obliczeniach

TFloatArray arrayError - macierz błędów

float b - wartość b

float bDiff - delta b

Metody klasy SVMClassifier:

SVMClassifier() - konstruktor, tworzy obiekt SVMClassifier

SVMClassifier() - destruktor

int train() - funkcja ucząca klasyfikatora, implementuje główną procedurę uczenia minimalną optymalizacją sekwencyjną, zapisuje rezultaty do pliku

int examineExample(int i1) - funkcja która przy wywołaniu otrzymuje indeks alphy, sprawdza warunki KKT (herustyka) oraz wyszukuje wartości alpha2, następnie wywołuje funkcję takeStep() - optymalizacja dwóch punktów

int takeStep(int i1, int i2) - optymalizacja dwóch mnożników Lagrange'a, zwraca 1 w przypadku udanej optymalizacji, 0 w przypadku nieudanej, i1, i2 - indeksy alph

int predict() - funkcja klasyfikująca próbki testowe

float errorRate() - funkcja zwracająca Error Rate;
int loadResults(std::ifstream is)- funkcja otwierająca/tworząca plik z rezultatami
void writeResultModel(std::ofstream os) - funkcja zapisująca rezultaty do pliku
float kernel(int i1, int i2) - funkcja quadratic kernel zwracająca wartość K
float learnedFnc(int k) - wywoływane po zoptymalizowaniu, wylicza wartość nauczonej funkcji w punkcie k

1.2 Another functions

typedef std::pair \langle int, float \rangle TVectorDim
typedef std::vector \langle TVectorDim \rangle TVector
typedef std::vector \langle TVector \rangle TVectorArray
int splitCSV(const TString& s, char c, TStringArray& v) - funkcja rozdzielająca plik .csv
int readSample(TString& s, TVector& x, float& y)- funkcja czytająca pojedynczą próbkę z pliku csv
int writeSample(TString& s, TVector& x, float& y) - funkcja zapisującą próbkę do macierzy z cechami oraz z etykietami klasy
int partReadSample(std::ifstream& is, TVectorArray& arrayX, TFloatArray& arrayY, int& n) - funkcja czytająca cały plik z próbkami
int partWriteSample(std::ofstream& os, TVectorArray& arrayX, TFloatArray& arrayY, int& n) - funkcja zapisująca do pliku
float dotProduct(const TVector& v1, const TVector& v2) - funkcja obliczająca iloczyn skalarny
TVector operator*(const TVector& v, float f) - przeciążenie operatora **TVector operator*(float f, const TVector& v)**- zwraca wektor $v*f$
TVector operator+(const TVector& v1, const TVector& v2) - przeciążenie operatora **+**