5000 images, 5000 sampling points (533 dog, 467 cat):

* 20 clusters:
  + SVM:
    - Train:

Optimization finished, #iter = 2313

nu = 0.867133

obj = -4184.042797, rho = -0.750133

nSV = 4340, nBSV = 4332

Total nSV = 4340

* + - Test:

Accuracy = 64% (640/1000) (classification)

* 33 clusters:
  + SVM:
    - Train:

optimization finished, #iter = 2293

nu = 0.869194

obj = -4120.922823, rho = -0.404564

nSV = 4350, nBSV = 4342

Total nSV = 4350

* + - Test:

Accuracy = 67.9% (679/1000) (classification)

* + Nearest Neighbor:
    - Test:

Accuracy = 77.1%

* 30 clusters:
  + SVM:
    - Train:

optimization finished, #iter = 2321

nu = 0.865867

obj = -4140.694516, rho = -0.276631

nSV = 4333, nBSV = 4324

Total nSV = 4333

* + - Test:

Accuracy = 67.1% (671/1000) (classification)

* + Nearest Neighbor:
    - Test:

Accuracy = 77.5%

* 35 clusters:
  + SVM:
    - Train:

optimization finished, #iter = 2332

nu = 0.871794

obj = -4131.325936, rho = -0.178506

nSV = 4363, nBSV = 4356

Total nSV = 4363

* + - Test:

Accuracy = 68.1% (681/1000) (classification)

* + Nearest Neighbor:
    - Test:

Accuracy = 75%

* 40 clusters:
  + SVM:
    - Train:

optimization finished, #iter = 2306

nu = 0.872573

obj = -4145.945144, rho = -0.245792

nSV = 4366, nBSV = 4358

Total nSV = 4366

* + - Test:

Accuracy = 67.2% (672/1000) (classification)

* + Nearest Neighbor:
    - Test:

Accuracy = 77.1%

* 45 clusters:
  + SVM:
    - Train:

optimization finished, #iter = 2323

nu = 0.885016

obj = -4183.343485, rho = -0.238244

nSV = 4428, nBSV = 4420

Total nSV = 4428

* + - Test:

Accuracy = 67.7% (677/1000) (classification)

* + Nearest Neighbor:
    - Test:

Accuracy – 77.6%

* 50 clusters:
  + SVM:
    - Train:

optimization finished, #iter = 2317

nu = 0.887676

obj = -4191.650433, rho = -0.213622

nSV = 4442, nBSV = 4435

Total nSV = 4442

* + - Test:

Accuracy = 66.4% (664/1000) (classification)

* + Nearest Neighbor:
    - Test:

Accuracy = 75.1%

* 80 clusters:
  + SVM:
    - Train:

optimization finished, #iter = 2399

nu = 0.911568

obj = -4291.918853, rho = -0.154950

nSV = 4563, nBSV = 4556

Total nSV = 4563

* + - Test:

Accuracy = 66.8% (668/1000) (classification)

* + Nearest Neighbor:
    - Test:

Accuracy = 76%

3000 samples:

* 35 clusters:
  + SVM:
    - Train:

optimization finished, #iter = 2307

nu = 0.867966

obj = -4128.243042, rho = -0.339744

nSV = 4342, nBSV = 4336

Total nSV = 4342

* + - Test:

Accuracy = 66.1% (661/1000) (classification)

* + Nearest Neighbor:
    - Test:

Accuracy = 75.8%

8000 samples:

* 35 clusters:
  + SVM:
    - Train:

optimization finished, #iter = 2303

nu = 0.864621

obj = -4103.760758, rho = -0.250569

nSV = 4328, nBSV = 4317

Total nSV = 4328

* + - Test:

Accuracy = 67.6% (676/1000) (classification)

* + Nearest Neighbor:
    - Test:

Accuracy = 76.6%

8500 samples:

* 35 clusters:
  + SVM:
    - Train:

optimization finished, #iter = 2314

nu = 0.866017

obj = -4104.516532, rho = -0.349574

nSV = 4338, nBSV = 4325

Total nSV = 4338

* + - Test:

Accuracy = 67.8% (678/1000) (classification)

* + Nearest Neighbor:
    - Test:

Accuracy = 75.8%

9000 samples:

* 35 clusters:
  + SVM:
    - Train:

optimization finished, #iter = 2300

nu = 0.871095

obj = -4136.518337, rho = -0.289267

nSV = 4360, nBSV = 4352

Total nSV = 4360

/optimization finished, #iter = 2283

nu = 0.865473

obj = -4108.360020, rho = -0.201610

nSV = 4330, nBSV = 4324

Total nSV = 4330

* + - Test:

Accuracy = 68.3% (683/1000) (classification)/

Accuracy = 67.1% (671/1000) (classification)

* + Nearest Neighbor:
    - Test:

Accuracy = 76%

10000 samples:

* 35 clusters:
  + SVM:
    - Train:

optimization finished, #iter = 2304

nu = 0.868452

obj = -4125.123307, rho = -0.330109

nSV = 4347, nBSV = 4338

Total nSV = 4347

* + - Test:

Accuracy = 67.4% (674/1000) (classification)

* + Nearest Neighbor:
    - Test:

Accuracy=74.7%

Random cluster centers:

* 30 clusters:
  + SVM:
    - Train:

optimization finished, #iter = 2326

nu = 0.884562

obj = -4239.142757, rho = -0.594626

nSV = 4425, nBSV = 4420

Total nSV = 4425

* + - Test:

Accuracy = 64.5% (645/1000) (classification)

* + Nearest Neighbor:
    - Test:

Accuracy = 75%

* 35 clusters:
  + SVM:
    - Train:

optimization finished, #iter = 2321

nu = 0.888706

obj = -4239.935608, rho = -0.455720

nSV = 4448, nBSV = 4441

Total nSV = 4448

* + - Test:

Accuracy = 63% (630/1000) (classification)

* + Nearest Neighbor:
    - Test:

Accuracy = 75.8%

* 40 clusters:
  + SVM:
    - Train:

optimization finished, #iter = 2315

nu = 0.873151

obj = -4180.494933, rho = -0.239258

nSV = 4369, nBSV = 4362

Total nSV = 4369

* + - Test:

Accuracy = 66% (660/1000) (classification)

* + Nearest Neighbor:
    - Test:

Accuracy = 78.1%

* 50 clusters:
  + SVM:
    - Train:

optimization finished, #iter = 2336

nu = 0.880853

obj = -4188.205345, rho = -0.039389

nSV = 4407, nBSV = 4401

Total nSV = 4407

* + - Test:

Accuracy = 66.7% (667/1000) (classification)

* + Nearest Neighbor:
    - Test:

Accuracy = 74.2%

Kernel- linear

model1 = svmtrain(result, bow,'-t 0');

Kernel- polynomial w/ degree 0

model2 = svmtrain(result, bow,'-t 1 -d 0');

Kernel- polynomial w/ degree 1

Model3 = svmtrain(result, bow,'-t 1 -d 1');

Kernel- polynomial w/ degree 2

Model4 = svmtrain(result, bow,'-t 1 -d 2');

Kernel- polynomial w/ degree 3

Model5 = svmtrain(result, bow,'-t 1 -d 3');

Kernel- polynomial w/ degree 100

model6 = svmtrain(result, bow,'-t 1 -d 100');

Kernel- polynomial w/ degree 0.5

model7 = svmtrain(result, bow,'-t 1 -d 0.5');

Kernel- polynomial w/ gamma 0.1

model8 = svmtrain(result, bow,'-t 1 -g 0.1');

Kernel- polynomial w/ gamma 0.5

model9 = svmtrain(result, bow,'-t 1 -g 0.5');

Kernel- polynomial w/ gamma 1

model10 = svmtrain(result, bow,'-t 1 -g 1');

Kernel- polynomial w/ gamma 10

model11 = svmtrain(result, bow,'-t 1 -g 10');

Kernel- RBF w/ gamma 0.1

model12 = svmtrain(result, bow,'-t 2 -g 0.1');

Kernel- RBF w/ gamma 0.5

model13 = svmtrain(result, bow,'-t 2 -g 0.5');

Kernel- RBF w/ gamma 1

model14 = svmtrain(result, bow,'-t 2 -g 1');

Kernel- RBF w/ gamma 10

model15 = svmtrain(result, bow,'-t 2 -g 10');

Kernel- RBF w/ gamma 50

model16 = svmtrain(result, bow,'-t 2 -g 50');

Kernel- RBF w/ gamma 500

model17 = svmtrain(result, bow,'-t 2 -g 500');

Kernel- sigmoid w/ gamma 0.1

model18 = svmtrain(result, bow,'-t 3 -g 0.1');

Kernel- sigmoid w/ gamma 0.5

model19 = svmtrain(result, bow,'-t 3 -g 0.5');

Kernel- sigmoid w/ gamma 1

Model20 = svmtrain(result, bow,'-t 3 -g 1');

Kernel- sigmoid w/ gamma 10

Model21 = svmtrain(result, bow,'-t 3 -g 10');

During model construction, I used the same set of training data for all model. Each model is slightly different than the others one since each was trained with different parameter passed in. First model (model1) is constructed with linear kernel function, which has an accuracy of 68.7%. For model2 through model 7, they are trained with polynomial kernel function and degree parameter being set in the range from 0-3, which operates at chance performance (52% to 53.5%). Since the default for the degree parameter is 3, I concluded that this kernel function does not fit this particular data set. However, as I set the degree parameter to 100 and used the polynomial kernel function, it yields a 99.3% accuracy, which contradicted my previous conclusion (model 8). Then instead of changing degree parameter, I changed the gamma parameter (default set to 1/number of clusters). Models are trained with polynomial kernel function as gamma being 0.1, 0.5,1,10. The first three models (model9-10) generate an accuracy closed to the first model, which is trained with linear kernel function (61.8, 68.7,70.8 respectfully). Again, the last model yields an accuracy of 91.11%. For the next six models, I set gamma parameter as 0.1,0.5, 1, 10, 50, and 500, and trained with radial basis function kernel type. The first three models yield accuracies at around 69%. For the fourth model (gamma = 10), the accuracy increases to 86.5. Surprisingly, the last two models (gamma = 50 and 500) generate accuracies of 100%. For the last four models I did, I trained the model with sigmoid kernel type with 0.1, 0.5, 1, and 10 as being gamma. The first two gave out accuracies of 67.5% and 68.2%, and the last two operates at chance performance.