**allwallfeatures\_pca300\_nov17.mtx"**

pca with 300 dimensions

ntrain = 250

nvalidation = 50

lambda\_list = list(range(5,150))

lasso

>>> test\_list

array([ 1., 1., 0., 1., 0., 1., 1., 1., 1., 0., 1., 1., 1.,

1., 1., 1., 1., 1., 1., 1., 0., 1., 1., 1., 1., 1.,

1., 1., 1., 1., 1., 1., 0., 0., 1., 0., 0., 1., 1.,

1., 0., 1., 1., 1., 1., 1., 1., 1., 1., 1., 0., 0.,

0., 1., 0., 1., 1., 1., 0., 1.])

>>> sum(test\_list)

46.0

>>> sum(test\_list) / test\_list.size

0.76666666666666672

>>> io.mmwrite("pca300", pca)

pca300 with lambda from 0 – 10

error on test set : 0.683333333333

pca with 100 components and lambda 50-150

words guessed correctly: 45.0

percentage guessed correctly: 0.75

[ 1. 1. 0. 1. 0. 0. 1. 1. 1. 0. 1. 1. 1. 1. 1. 1. 1. 1.

1. 1. 0. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 0. 0. 1. 0.

0. 1. 1. 1. 0. 1. 1. 1. 1. 1. 1. 1. 1. 1. 0. 0. 0. 1.

0. 1. 1. 1. 0. 1.]

pca with 50 dimensions

words guessed correctly: 44.0

percentage guessed correctly: 0.733333333333

lambda 50-100

dimensions – 300

accuracy improves dramatically at ~12 features

we get a max accuracy of 0.8 at about 120 features

array([ 0.26666667, 0.28333333, 0.36666667, 0.38333333, 0.43333333,

0.46666667, 0.48333333, 0.5 , 0.53333333, 0.58333333,

0.61666667, 0.71666667, 0.71666667, 0.68333333, 0.68333333,

0.7 , 0.66666667, 0.68333333, 0.68333333, 0.68333333,

0.68333333, 0.66666667, 0.65 , 0.66666667, 0.66666667,

0.66666667, 0.66666667, 0.68333333, 0.68333333, 0.66666667,

0.65 , 0.66666667, 0.68333333, 0.7 , 0.7 ,

0.68333333, 0.68333333, 0.7 , 0.7 , 0.7 ,

0.7 , 0.68333333, 0.68333333, 0.7 , 0.7 ,

0.7 , 0.66666667, 0.68333333, 0.68333333, 0.68333333,

0.68333333, 0.66666667, 0.68333333, 0.68333333, 0.7 ,

0.71666667, 0.7 , 0.7 , 0.68333333, 0.68333333,

0.7 , 0.68333333, 0.7 , 0.7 , 0.71666667,

0.73333333, 0.71666667, 0.75 , 0.73333333, 0.73333333,

0.71666667, 0.73333333, 0.73333333, 0.73333333, 0.73333333,

0.73333333, 0.73333333, 0.71666667, 0.71666667, 0.7 ,

0.71666667, 0.71666667, 0.71666667, 0.71666667, 0.71666667,

0.71666667, 0.71666667, 0.73333333, 0.73333333, 0.73333333,

0.73333333, 0.73333333, 0.73333333, 0.71666667, 0.71666667,

0.71666667, 0.71666667, 0.71666667, 0.75 , 0.75 ,

0.75 , 0.75 , 0.75 , 0.75 , 0.75 ,

0.75 , 0.73333333, 0.73333333, 0.75 , 0.73333333,

0.75 , 0.76666667, 0.76666667, 0.76666667, 0.76666667,

0.76666667, 0.78333333, 0.8 , 0.76666667, 0.78333333,

0.78333333, 0.78333333, 0.78333333, 0.78333333, 0.78333333,

0.78333333, 0.78333333, 0.78333333, 0.78333333, 0.78333333,

0.78333333, 0.78333333, 0.78333333, 0.78333333, 0.76666667,

0.75 , 0.75 , 0.75 , 0.76666667, 0.76666667,

0.76666667, 0.75 , 0.75 , 0.78333333, 0.76666667,

0.76666667, 0.76666667, 0.73333333, 0.75 , 0.76666667,

0.76666667, 0.76666667, 0.75 , 0.76666667, 0.75 ,

0.76666667, 0.76666667, 0.76666667, 0.76666667, 0.76666667,

0.76666667, 0.76666667, 0.75 , 0.75 , 0.75 ,

0.73333333, 0.73333333, 0.73333333, 0.75 , 0.75 ,

0.75 , 0.75 , 0.75 , 0.75 , 0.75 ,

0.75 , 0.75 , 0.75 , 0.75 , 0.75 ,

0.75 , 0.75 , 0.75 , 0.75 , 0.75 ,

0.75 , 0.75 , 0.75 , 0.75 , 0.75 ,

0.75 , 0.73333333, 0.73333333, 0.73333333, 0.73333333,

0.73333333, 0.73333333, 0.73333333, 0.73333333, 0.75 ,

0.75 , 0.75 , 0.75 , 0.75 , 0.75 ,

0.75 , 0.75 , 0.75 , 0.75 , 0.75 ,

0.75 , 0.75 , 0.75 , 0.75 , 0.75 ,

0.75 , 0.75 , 0.75 ])