

a01_5

October 3, 2025

1 5 Generating Data

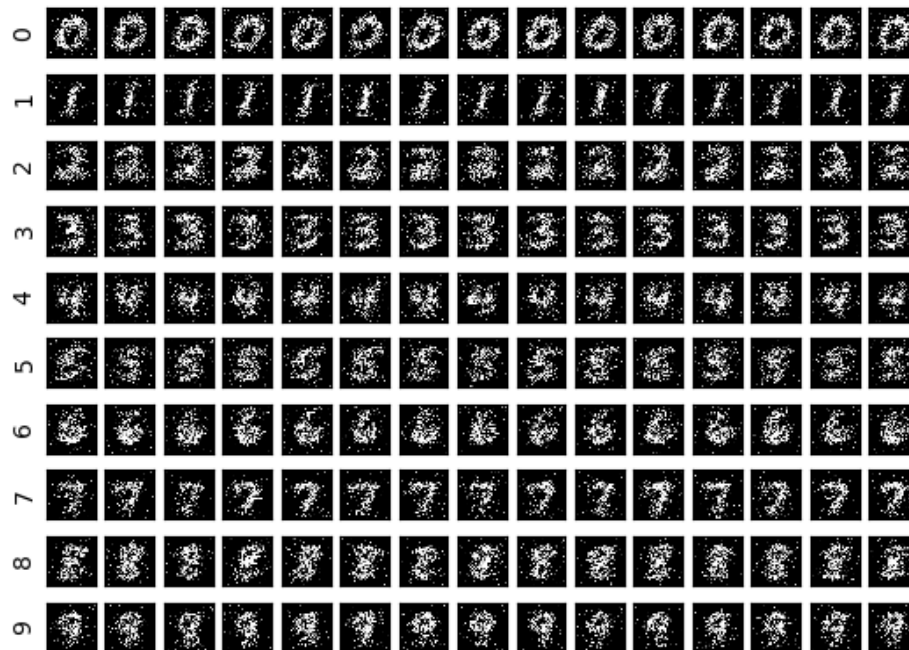
```
[ ]: %load_ext autoreload
      %autoreload 2

      from a01_helper import *
      from a01_functions import nb_train, nb_generate

[ ]: # let's generate 15 digits from each class and plot

      model_nb2 = nb_train(X, y, alpha=2)
      ygen = np.repeat(np.arange(10), 15)
      Xgen = nb_generate(model_nb2, ygen)

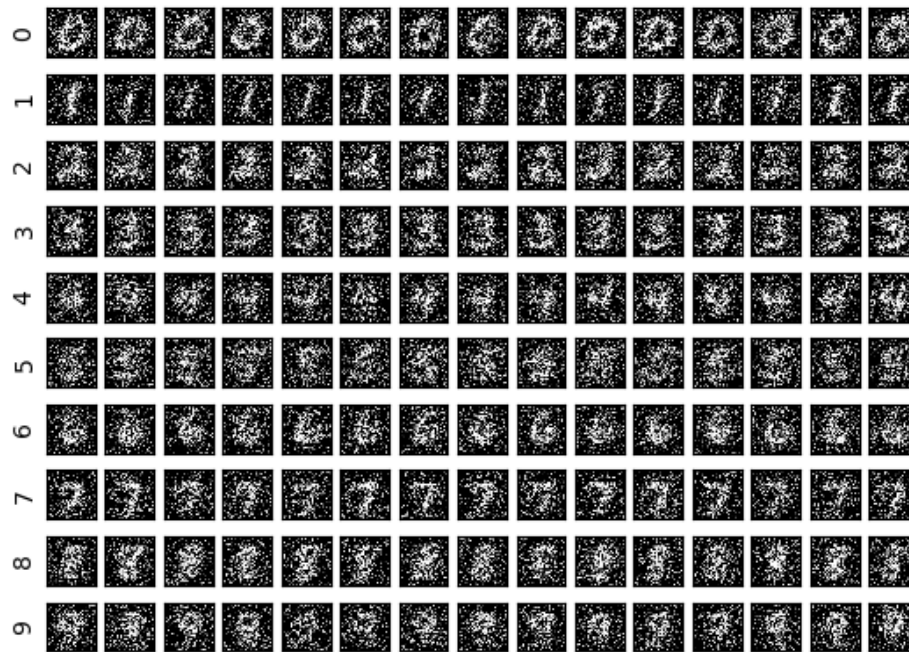
      nextplot()
      showdigits(Xgen, ygen)
      plt.savefig("images/task_5/nb_sampling_alpha_2", dpi=300)
```



```
[ ]: # let's generate 15 digits from each class and plot

model_nb2 = nb_train(X, y, alpha=10)
ygen = np.repeat(np.arange(10), 15)
Xgen = nb_generate(model_nb2, ygen)

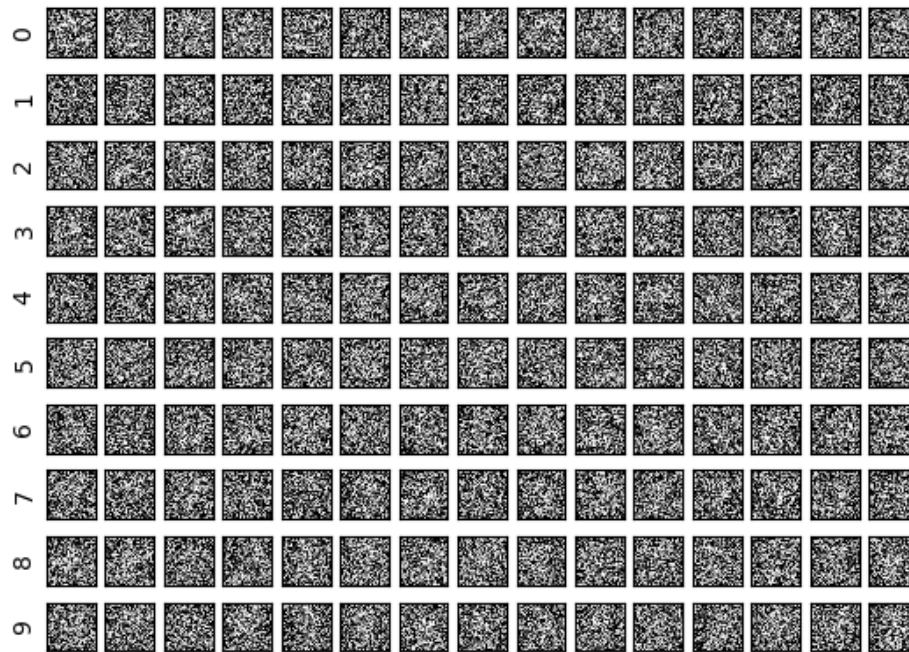
nextplot()
showdigits(Xgen, ygen)
plt.savefig("images/task_5/nb_sampling_alpha_10", dpi=300)
```



```
[ ]: # let's generate 15 digits from each class and plot

model_nb2 = nb_train(X, y, alpha=100)
ygen = np.repeat(np.arange(10), 15)
Xgen = nb_generate(model_nb2, ygen)

nextplot()
showdigits(Xgen, ygen)
plt.savefig("images/task_5/nb_sampling_alpha_100", dpi=300)
```



```
[ ]: # we can also plot the parameter vectors by choosing the most-likely
      # value for each feature
      ymax = np.arange(10)
      Xmax = np.zeros((10, D))
      for c in range(10):
          Xmax[c,] = np.apply_along_axis(np.argmax, 1, model_nb2["logcls"][c, :, :])

      nextplot()
      showdigits(Xmax, ymax)
      plt.savefig('images/task_5/nb_feature_vector', dpi=300)
```



```
[ ]: # Or the expected value of each feature. Here we leave the categorical domain
# and treat each feature as a number, i.e., this is NOT how categorical Naive
# Bayes sees it and we wouldn't be able to do this if the data were really
# categorical.
ymean = np.arange(10)
Xmean = np.zeros((10, D))
for c in range(10):
    Xmean[c,] = np.apply_along_axis(
        np.sum, 1, np.exp(model_nb2["logcls"][c, :, :]) * np.arange(256)
    )

nextplot()
showdigits(Xmean, ymean)
plt.savefig('images/task_5/nb_expected_values', dpi=300)
```

[]:

9	8	7	6	5	4	3	2	1	0
9	8	7	6	5	4	3	2	1	0