mnist_noise-current

October 6, 2020

[ ]:
```
'''
VARIABLE: METHOD OF DR

CONSTANT: N_EPOCHS, N_COMPONENTS, NOISE TYPE AND MAGNITUDE

'''
```

[38]:
```
# -- IMPORTS -- #
import math
import scipy
import numpy as np
from skimage import util
import matplotlib.pyplot as plt
from keras.datasets import mnist
from keras.utils import to_categorical
```

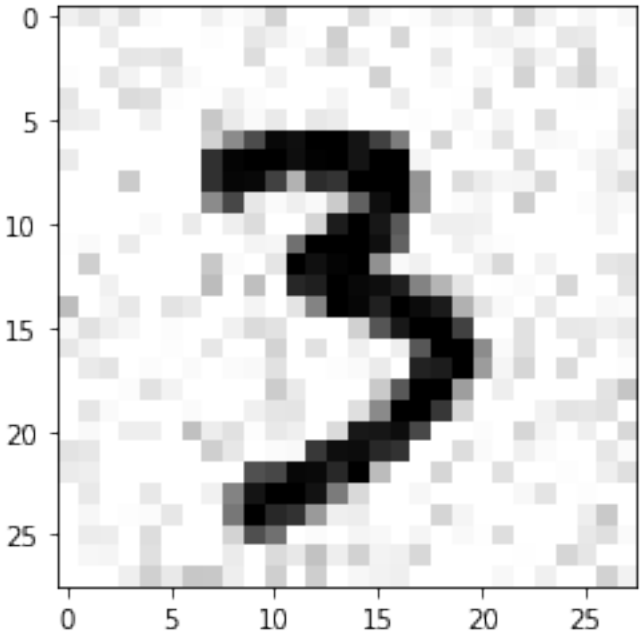
[39]:
```
# -- TRAIN AND TEST DATA PREPARATION -- #
(x_train, y_train), (x_test, y_test) = mnist.load_data()
x_test_noisy = np.empty((len(x_test),28,28),dtype="float64")

std = 0.1
for i in range(len(x_test)):
    x_test_noisy[i] = util.random_noise(x_test[i],mode="gaussian",var=std**2)
```

[40]:
```
# -- OUTPUT SNR AND SAMPLE NOISY IMAGE -- #
p_signal = np.mean(x_test[30])
p_noise = std
snr = 10 * math.log(p_signal/p_noise,10)
print(snr)

plt.imshow(x_test_noisy[30],cmap=plt.cm.binary)
plt.show()
```

```
25.023887949131005
```



[41]:
```
# -- CONVERTING DATA FOR INPUT INTO NEURAL NETWORK -- #
x_train = x_train.reshape((60000,28*28))
x_train = x_train.astype("float32") / 255

x_test = x_test.reshape((10000,28*28))
x_test_noisy = x_test_noisy.reshape((10000,28*28))

y_train = to_categorical(y_train)
y_test = to_categorical(y_test)
```

[42]:
```
# -- DIMENSIONALITY REDUCTION -- #
import dim_reduction
x_train0, x_test_noisy0 = x_train.copy(), x_test_noisy.copy()
x_train1, x_test_noisy1 = dim_reduction.apply(30, x_train, x_test_noisy, "PCA")
x_train2, x_test_noisy2 = dim_reduction.apply(30, x_train, x_test_noisy, "FA")
x_train3, x_test_noisy3 = dim_reduction.apply(30, x_train, x_test_noisy, "ICA")
```

```
C:\Users\Carl\.conda\envs\tf\lib\site-
packages\sklearn\decomposition\_fastica.py:120: ConvergenceWarning: FastICA did
not converge. Consider increasing tolerance or the maximum number of iterations.
  ConvergenceWarning)
```

[43]:
```
# -- NEURAL NETWORK -- #
import neural_network
history0 = neural_network.fit(784, 30, x_train0, y_train, x_test_noisy0, y_test)
history1 = neural_network.fit(30, 30, x_train1, y_train, x_test_noisy1, y_test)
```

```
history2 = neural_network.fit(30, 30, x_train2, y_train, x_test_noisy2, y_test)
history3 = neural_network.fit(30, 30, x_train3, y_train, x_test_noisy3, y_test)
```

```
Train on 60000 samples, validate on 10000 samples
Epoch 1/30
60000/60000 [==============================] - 1s 12us/step - loss: 1.5624 -
accuracy: 0.6049 - val_loss: 1.0544 - val_accuracy: 0.8020
Epoch 2/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.7651 -
accuracy: 0.8381 - val_loss: 0.6045 - val_accuracy: 0.8701
Epoch 3/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.5002 -
accuracy: 0.8768 - val_loss: 0.4551 - val_accuracy: 0.8918
Epoch 4/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.4007 -
accuracy: 0.8949 - val_loss: 0.3901 - val_accuracy: 0.9029
Epoch 5/30
60000/60000 [==============================] - 1s 13us/step - loss: 0.3505 -
accuracy: 0.9055 - val_loss: 0.3533 - val_accuracy: 0.9092
Epoch 6/30
60000/60000 [==============================] - 1s 13us/step - loss: 0.3196 -
accuracy: 0.9126 - val_loss: 0.3325 - val_accuracy: 0.9132
Epoch 7/30
60000/60000 [==============================] - 1s 14us/step - loss: 0.2978 -
accuracy: 0.9182 - val_loss: 0.3137 - val_accuracy: 0.9174
Epoch 8/30
60000/60000 [==============================] - 1s 14us/step - loss: 0.2809 -
accuracy: 0.9225 - val_loss: 0.3047 - val_accuracy: 0.9195
Epoch 9/30
60000/60000 [==============================] - 1s 14us/step - loss: 0.2669 -
accuracy: 0.9265 - val_loss: 0.2912 - val_accuracy: 0.9235
Epoch 10/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.2548 -
accuracy: 0.9302 - val_loss: 0.2863 - val_accuracy: 0.9250
Epoch 11/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.2443 -
accuracy: 0.9325 - val_loss: 0.2784 - val_accuracy: 0.9260
Epoch 12/30
60000/60000 [==============================] - 1s 13us/step - loss: 0.2349 -
accuracy: 0.9350 - val_loss: 0.2678 - val_accuracy: 0.9284
Epoch 13/30
60000/60000 [==============================] - 1s 13us/step - loss: 0.2264 -
accuracy: 0.9372 - val_loss: 0.2631 - val_accuracy: 0.9304
Epoch 14/30
60000/60000 [==============================] - 1s 13us/step - loss: 0.2185 -
accuracy: 0.9395 - val_loss: 0.2580 - val_accuracy: 0.9317
Epoch 15/30
```

```
60000/60000 [==============================] - 1s 13us/step - loss: 0.2114 -
accuracy: 0.9419 - val_loss: 0.2543 - val_accuracy: 0.9309
Epoch 16/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.2047 -
accuracy: 0.9432 - val_loss: 0.2511 - val_accuracy: 0.9319
Epoch 17/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.1986 -
accuracy: 0.9452 - val_loss: 0.2467 - val_accuracy: 0.9336
Epoch 18/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.1928 -
accuracy: 0.9463 - val_loss: 0.2442 - val_accuracy: 0.9347
Epoch 19/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.1874 -
accuracy: 0.9478 - val_loss: 0.2388 - val_accuracy: 0.9347
Epoch 20/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.1823 -
accuracy: 0.9493 - val_loss: 0.2353 - val_accuracy: 0.9357
Epoch 21/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.1773 -
accuracy: 0.9503 - val_loss: 0.2351 - val_accuracy: 0.9363
Epoch 22/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.1727 -
accuracy: 0.9518 - val_loss: 0.2254 - val_accuracy: 0.9398
Epoch 23/30
60000/60000 [==============================] - 1s 13us/step - loss: 0.1683 -
accuracy: 0.9529 - val_loss: 0.2274 - val_accuracy: 0.9381
Epoch 24/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.1641 -
accuracy: 0.9538 - val_loss: 0.2224 - val_accuracy: 0.9392
Epoch 25/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.1602 -
accuracy: 0.9553 - val_loss: 0.2151 - val_accuracy: 0.9423
Epoch 26/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.1563 -
accuracy: 0.9563 - val_loss: 0.2107 - val_accuracy: 0.9442
Epoch 27/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.1528 -
accuracy: 0.9575 - val_loss: 0.2140 - val_accuracy: 0.9435
Epoch 28/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.1492 -
accuracy: 0.9584 - val_loss: 0.2183 - val_accuracy: 0.9400
Epoch 29/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.1459 -
accuracy: 0.9594 - val_loss: 0.2162 - val_accuracy: 0.9406
Epoch 30/30
60000/60000 [==============================] - 1s 12us/step - loss: 0.1427 -
accuracy: 0.9603 - val_loss: 0.2017 - val_accuracy: 0.9467
Train on 60000 samples, validate on 10000 samples
```

```
Epoch 1/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.2537 -
accuracy: 0.2604 - val_loss: 4.1049 - val_accuracy: 0.2098
Epoch 2/30
60000/60000 [==============================] - 0s 3us/step - loss: 1.2629 -
accuracy: 0.6190 - val_loss: 3.0636 - val_accuracy: 0.2991
Epoch 3/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.7953 -
accuracy: 0.7764 - val_loss: 2.5292 - val_accuracy: 0.3937
Epoch 4/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.5826 -
accuracy: 0.8339 - val_loss: 2.4302 - val_accuracy: 0.4272
Epoch 5/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.4740 -
accuracy: 0.8640 - val_loss: 2.4965 - val_accuracy: 0.4428
Epoch 6/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.4108 -
accuracy: 0.8811 - val_loss: 2.6164 - val_accuracy: 0.4457
Epoch 7/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.3699 -
accuracy: 0.8927 - val_loss: 2.7772 - val_accuracy: 0.4374
Epoch 8/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.3410 -
accuracy: 0.9005 - val_loss: 2.8633 - val_accuracy: 0.4358
Epoch 9/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.3192 -
accuracy: 0.9060 - val_loss: 2.9916 - val_accuracy: 0.4310
Epoch 10/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.3018 -
accuracy: 0.9111 - val_loss: 3.0738 - val_accuracy: 0.4275
Epoch 11/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2875 -
accuracy: 0.9154 - val_loss: 3.2028 - val_accuracy: 0.4198
Epoch 12/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2754 -
accuracy: 0.9186 - val_loss: 3.2458 - val_accuracy: 0.4197
Epoch 13/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2650 -
accuracy: 0.9215 - val_loss: 3.2611 - val_accuracy: 0.4209
Epoch 14/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2556 -
accuracy: 0.9239 - val_loss: 3.3425 - val_accuracy: 0.4168
Epoch 15/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2475 -
accuracy: 0.9269 - val_loss: 3.3402 - val_accuracy: 0.4209
Epoch 16/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2400 -
accuracy: 0.9283 - val_loss: 3.3541 - val_accuracy: 0.4218
```

```
Epoch 17/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2332 -
accuracy: 0.9306 - val_loss: 3.4020 - val_accuracy: 0.4198
Epoch 18/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2269 -
accuracy: 0.9327 - val_loss: 3.4402 - val_accuracy: 0.4193
Epoch 19/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2211 -
accuracy: 0.9345 - val_loss: 3.4570 - val_accuracy: 0.4198
Epoch 20/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2159 -
accuracy: 0.9359 - val_loss: 3.5377 - val_accuracy: 0.4160
Epoch 21/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2109 -
accuracy: 0.9374 - val_loss: 3.5544 - val_accuracy: 0.4163
Epoch 22/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2063 -
accuracy: 0.9388 - val_loss: 3.5425 - val_accuracy: 0.4186
Epoch 23/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2019 -
accuracy: 0.9401 - val_loss: 3.5231 - val_accuracy: 0.4233
Epoch 24/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.1978 -
accuracy: 0.9413 - val_loss: 3.5776 - val_accuracy: 0.4220
Epoch 25/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.1940 -
accuracy: 0.9425 - val_loss: 3.5661 - val_accuracy: 0.4234
Epoch 26/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.1905 -
accuracy: 0.9437 - val_loss: 3.6062 - val_accuracy: 0.4232
Epoch 27/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.1870 -
accuracy: 0.9446 - val_loss: 3.5944 - val_accuracy: 0.4252
Epoch 28/30
60000/60000 [==============================] - 0s 3us/step - loss: 0.1838 -
accuracy: 0.9455 - val_loss: 3.6442 - val_accuracy: 0.4225
Epoch 29/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.1808 -
accuracy: 0.9463 - val_loss: 3.6453 - val_accuracy: 0.4257
Epoch 30/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.1778 -
accuracy: 0.9473 - val_loss: 3.6676 - val_accuracy: 0.4251
Train on 60000 samples, validate on 10000 samples
Epoch 1/30
60000/60000 [==============================] - 0s 5us/step - loss: 2.2595 -
accuracy: 0.1711 - val_loss: 2.1160 - val_accuracy: 0.2938
Epoch 2/30
60000/60000 [==============================] - 0s 4us/step - loss: 1.8935 -
```

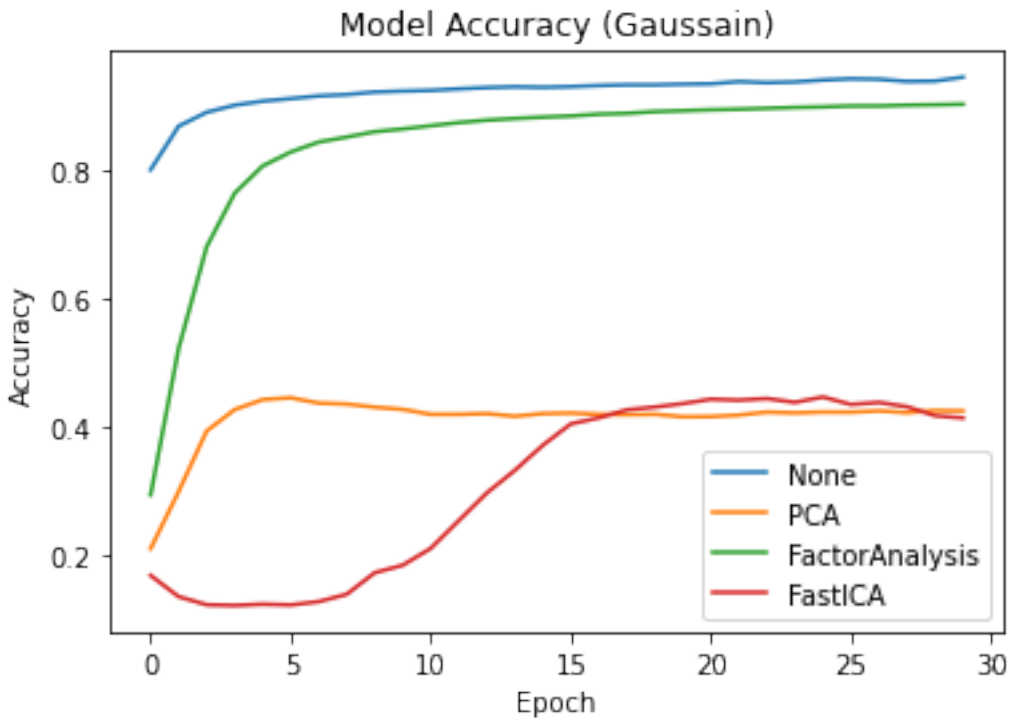
```
accuracy: 0.4705 - val_loss: 1.8019 - val_accuracy: 0.5228
Epoch 3/30
60000/60000 [==============================] - 0s 4us/step - loss: 1.5775 -
accuracy: 0.6872 - val_loss: 1.5257 - val_accuracy: 0.6814
Epoch 4/30
60000/60000 [==============================] - 0s 4us/step - loss: 1.3044 -
accuracy: 0.7799 - val_loss: 1.2837 - val_accuracy: 0.7654
Epoch 5/30
60000/60000 [==============================] - 0s 4us/step - loss: 1.0745 -
accuracy: 0.8217 - val_loss: 1.0801 - val_accuracy: 0.8079
Epoch 6/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.8881 -
accuracy: 0.8443 - val_loss: 0.9136 - val_accuracy: 0.8297
Epoch 7/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.7424 -
accuracy: 0.8583 - val_loss: 0.7832 - val_accuracy: 0.8453
Epoch 8/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.6315 -
accuracy: 0.8698 - val_loss: 0.6839 - val_accuracy: 0.8531
Epoch 9/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.5487 -
accuracy: 0.8781 - val_loss: 0.6098 - val_accuracy: 0.8615
Epoch 10/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.4870 -
accuracy: 0.8847 - val_loss: 0.5542 - val_accuracy: 0.8657
Epoch 11/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.4407 -
accuracy: 0.8906 - val_loss: 0.5123 - val_accuracy: 0.8708
Epoch 12/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.4052 -
accuracy: 0.8961 - val_loss: 0.4802 - val_accuracy: 0.8757
Epoch 13/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.3775 -
accuracy: 0.9003 - val_loss: 0.4550 - val_accuracy: 0.8795
Epoch 14/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.3557 -
accuracy: 0.9043 - val_loss: 0.4354 - val_accuracy: 0.8821
Epoch 15/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.3379 -
accuracy: 0.9074 - val_loss: 0.4203 - val_accuracy: 0.8842
Epoch 16/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.3233 -
accuracy: 0.9103 - val_loss: 0.4063 - val_accuracy: 0.8860
Epoch 17/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.3107 -
accuracy: 0.9128 - val_loss: 0.3950 - val_accuracy: 0.8889
Epoch 18/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2999 -
```

```
accuracy: 0.9153 - val_loss: 0.3853 - val_accuracy: 0.8901
Epoch 19/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2902 -
accuracy: 0.9171 - val_loss: 0.3773 - val_accuracy: 0.8929
Epoch 20/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2817 -
accuracy: 0.9191 - val_loss: 0.3691 - val_accuracy: 0.8943
Epoch 21/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2740 -
accuracy: 0.9211 - val_loss: 0.3628 - val_accuracy: 0.8957
Epoch 22/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2670 -
accuracy: 0.9229 - val_loss: 0.3557 - val_accuracy: 0.8966
Epoch 23/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2606 -
accuracy: 0.9246 - val_loss: 0.3512 - val_accuracy: 0.8980
Epoch 24/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2546 -
accuracy: 0.9263 - val_loss: 0.3461 - val_accuracy: 0.8994
Epoch 25/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2490 -
accuracy: 0.9280 - val_loss: 0.3414 - val_accuracy: 0.9003
Epoch 26/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2438 -
accuracy: 0.9294 - val_loss: 0.3360 - val_accuracy: 0.9017
Epoch 27/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2389 -
accuracy: 0.9306 - val_loss: 0.3329 - val_accuracy: 0.9017
Epoch 28/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2343 -
accuracy: 0.9319 - val_loss: 0.3283 - val_accuracy: 0.9029
Epoch 29/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2299 -
accuracy: 0.9332 - val_loss: 0.3242 - val_accuracy: 0.9035
Epoch 30/30
60000/60000 [==============================] - 0s 4us/step - loss: 0.2258 -
accuracy: 0.9342 - val_loss: 0.3204 - val_accuracy: 0.9044
Train on 60000 samples, validate on 10000 samples
Epoch 1/30
60000/60000 [==============================] - 0s 5us/step - loss: 2.3002 -
accuracy: 0.2233 - val_loss: 2.3001 - val_accuracy: 0.1681
Epoch 2/30
60000/60000 [==============================] - 0s 5us/step - loss: 2.2978 -
accuracy: 0.1321 - val_loss: 2.2980 - val_accuracy: 0.1349
Epoch 3/30
60000/60000 [==============================] - 0s 5us/step - loss: 2.2954 -
accuracy: 0.1226 - val_loss: 2.2957 - val_accuracy: 0.1222
Epoch 4/30
```

```
60000/60000 [==============================] - 0s 5us/step - loss: 2.2928 -
accuracy: 0.1198 - val_loss: 2.2934 - val_accuracy: 0.1211
Epoch 5/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.2901 -
accuracy: 0.1188 - val_loss: 2.2908 - val_accuracy: 0.1231
Epoch 6/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.2872 -
accuracy: 0.1239 - val_loss: 2.2882 - val_accuracy: 0.1219
Epoch 7/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.2841 -
accuracy: 0.1246 - val_loss: 2.2852 - val_accuracy: 0.1271
Epoch 8/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.2807 -
accuracy: 0.1274 - val_loss: 2.2820 - val_accuracy: 0.1386
Epoch 9/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.2771 -
accuracy: 0.1284 - val_loss: 2.2786 - val_accuracy: 0.1723
Epoch 10/30
60000/60000 [==============================] - ETA: 0s - loss: 2.2735 -
accuracy: 0.15 - 0s 5us/step - loss: 2.2732 - accuracy: 0.1535 - val_loss:
2.2749 - val_accuracy: 0.1840
Epoch 11/30
60000/60000 [==============================] - 0s 5us/step - loss: 2.2691 -
accuracy: 0.1624 - val_loss: 2.2711 - val_accuracy: 0.2103
Epoch 12/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.2647 -
accuracy: 0.1804 - val_loss: 2.2670 - val_accuracy: 0.2537
Epoch 13/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.2600 -
accuracy: 0.2024 - val_loss: 2.2626 - val_accuracy: 0.2969
Epoch 14/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.2550 -
accuracy: 0.2393 - val_loss: 2.2579 - val_accuracy: 0.3322
Epoch 15/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.2497 -
accuracy: 0.2578 - val_loss: 2.2530 - val_accuracy: 0.3715
Epoch 16/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.2441 -
accuracy: 0.2997 - val_loss: 2.2478 - val_accuracy: 0.4049
Epoch 17/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.2381 -
accuracy: 0.3421 - val_loss: 2.2424 - val_accuracy: 0.4143
Epoch 18/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.2319 -
accuracy: 0.3708 - val_loss: 2.2368 - val_accuracy: 0.4269
Epoch 19/30
60000/60000 [==============================] - 0s 5us/step - loss: 2.2254 -
accuracy: 0.4000 - val_loss: 2.2311 - val_accuracy: 0.4311
```

```
Epoch 20/30
60000/60000 [==============================] - 0s 5us/step - loss: 2.2185 -
accuracy: 0.4312 - val_loss: 2.2249 - val_accuracy: 0.4366
Epoch 21/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.2113 -
accuracy: 0.4577 - val_loss: 2.2186 - val_accuracy: 0.4434
Epoch 22/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.2039 -
accuracy: 0.4882 - val_loss: 2.2121 - val_accuracy: 0.4420
Epoch 23/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.1961 -
accuracy: 0.5302 - val_loss: 2.2052 - val_accuracy: 0.4444
Epoch 24/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.1879 -
accuracy: 0.5338 - val_loss: 2.1981 - val_accuracy: 0.4385
Epoch 25/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.1796 -
accuracy: 0.5718 - val_loss: 2.1909 - val_accuracy: 0.4469
Epoch 26/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.1708 -
accuracy: 0.5894 - val_loss: 2.1837 - val_accuracy: 0.4352
Epoch 27/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.1618 -
accuracy: 0.6115 - val_loss: 2.1761 - val_accuracy: 0.4384
Epoch 28/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.1525 -
accuracy: 0.6251 - val_loss: 2.1681 - val_accuracy: 0.4317
Epoch 29/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.1429 -
accuracy: 0.6275 - val_loss: 2.1598 - val_accuracy: 0.4177
Epoch 30/30
60000/60000 [==============================] - 0s 4us/step - loss: 2.1329 -
accuracy: 0.6518 - val_loss: 2.1520 - val_accuracy: 0.4142
```

[44]:
```
# -- OUTPUT PLOTS -- #
plt.plot(history0.history["val_accuracy"])
plt.plot(history1.history["val_accuracy"])
plt.plot(history2.history["val_accuracy"])
plt.plot(history3.history["val_accuracy"])
plt.title("Model Accuracy (Gaussain)")
plt.ylabel("Accuracy")
plt.xlabel("Epoch")
plt.legend(["None", "PCA", "FactorAnalysis", "FastICA"],loc="lower right")
plt.show()
```


Model Accuracy (Gaussain)
Accuracy
0.2
0.4
0.6
0.8
0
5
10
15
20
25
30
Epoch
None
PCA
FactorAnalysis
FastICA