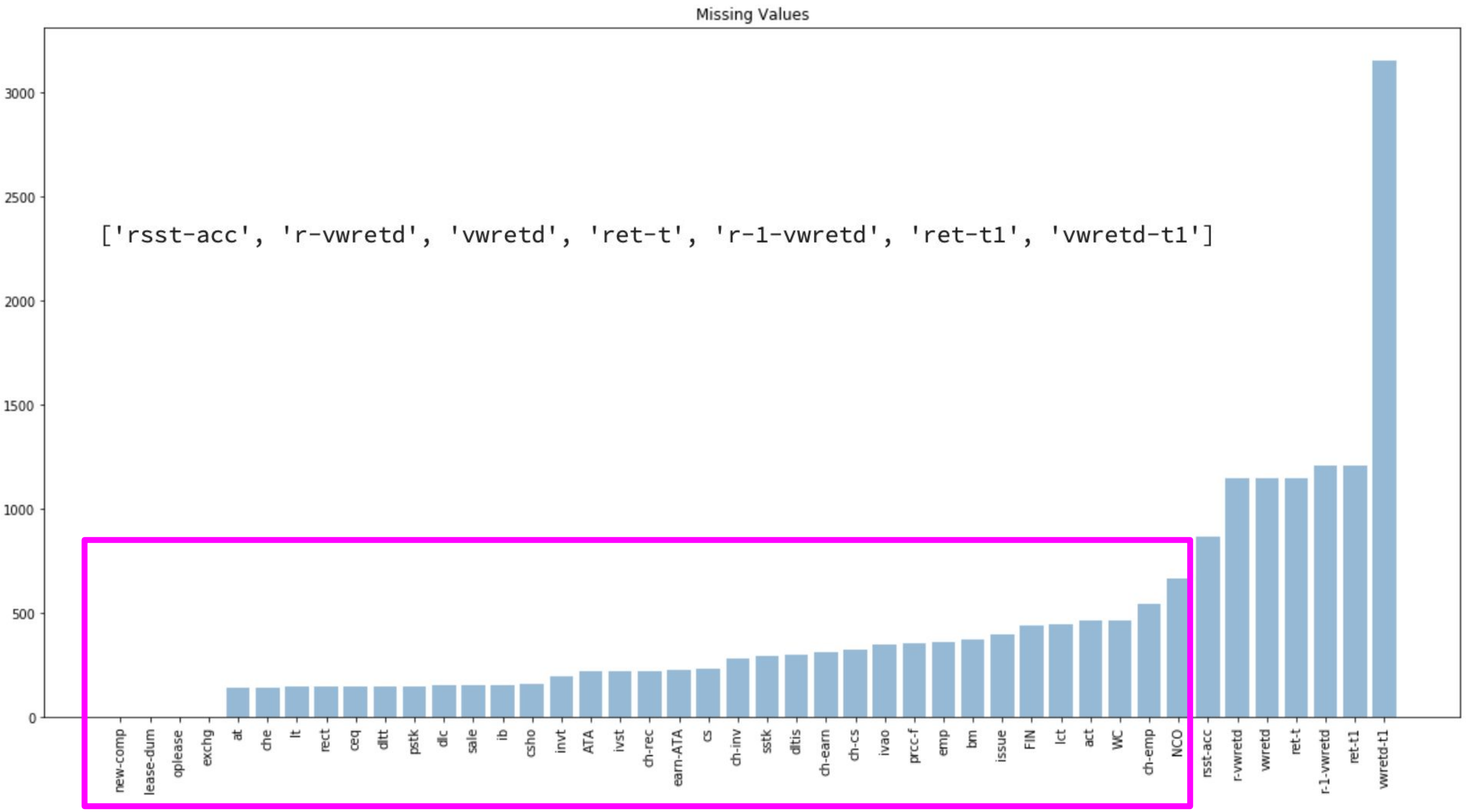
Dataset A + SdA

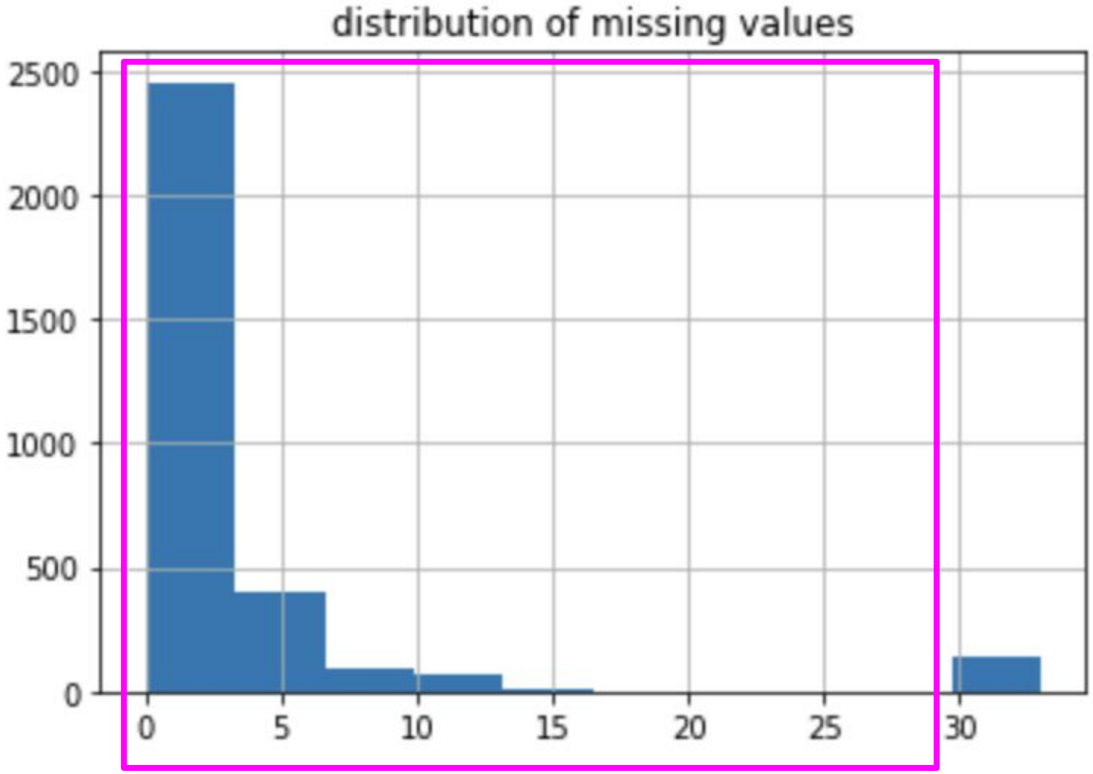
Hannah LeBlanc CISC873

Missing Values by Column



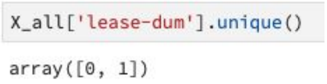
```
['rsst-acc', 'r-vwretd', 'vwretd', 'ret-t', 'r-1-vwretd', 'ret-t1', 'vwretd-t1']
```

Missing Values by Row

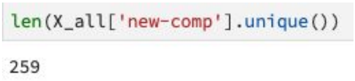


Data size after removing rows and columns

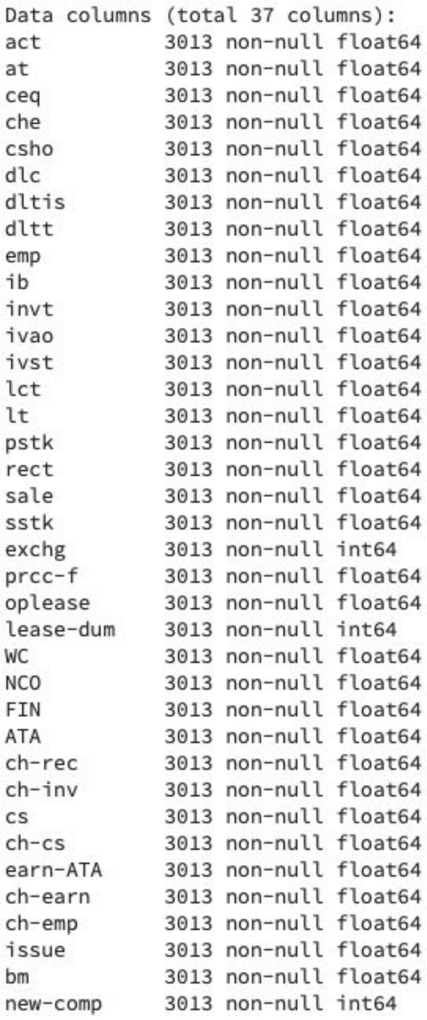
```
X shape:  (3013, 37)
y shape:  (3013, 1)
```



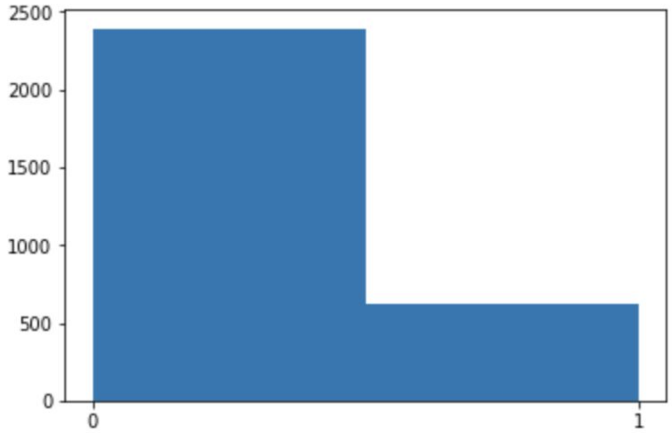
```
X_all['lease-dum'].unique()

array([0, 1])
```



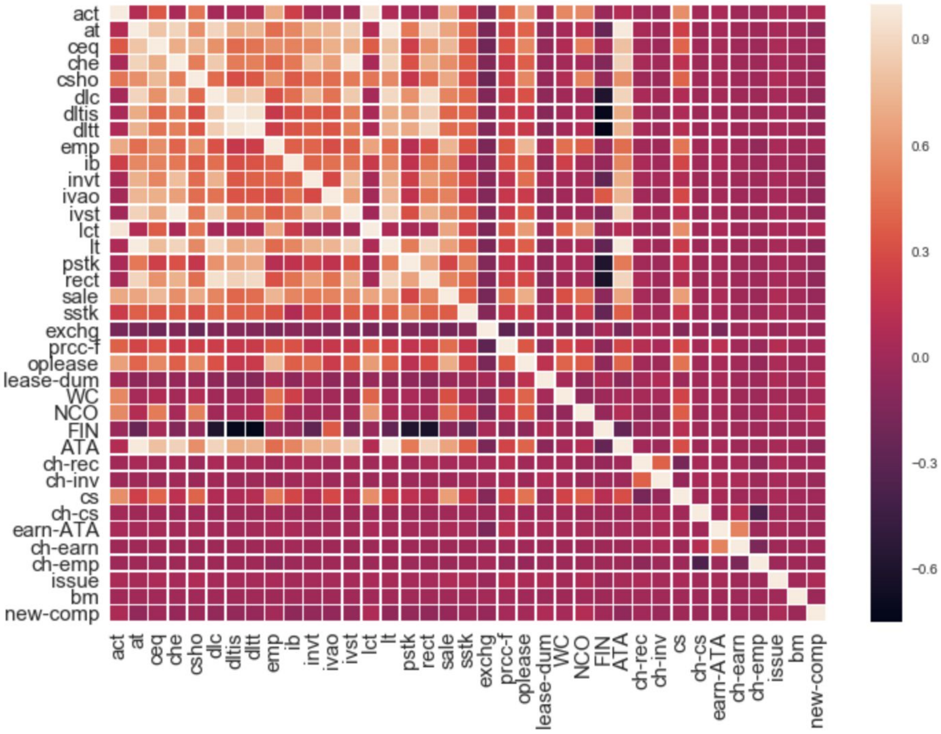
```
len(X_all['new-comp'].unique())

259
```



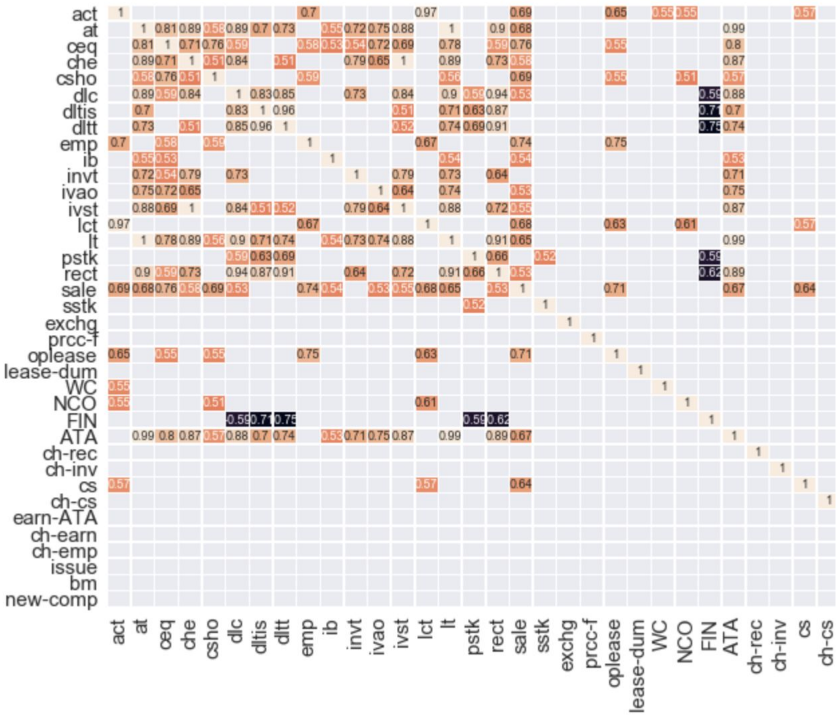
```
Data columns (total 37 columns):
act          3013 non-null float64
at           3013 non-null float64
ceq          3013 non-null float64
che          3013 non-null float64
csho         3013 non-null float64
dlc          3013 non-null float64
dltis        3013 non-null float64
dltt         3013 non-null float64
emp          3013 non-null float64
ib           3013 non-null float64
invt         3013 non-null float64
ivao         3013 non-null float64
ivst         3013 non-null float64
lct          3013 non-null float64
lt           3013 non-null float64
pstk         3013 non-null float64
rect         3013 non-null float64
sale         3013 non-null float64
sstk         3013 non-null float64
exchg        3013 non-null int64
prcc-f       3013 non-null float64
oplease      3013 non-null float64
lease-dum    3013 non-null int64
WC           3013 non-null float64
NCO          3013 non-null float64
FIN          3013 non-null float64
ATA          3013 non-null float64
ch-rec       3013 non-null float64
ch-inv       3013 non-null float64
cs           3013 non-null float64
ch-cs        3013 non-null float64
earn-ATA     3013 non-null float64
ch-earn      3013 non-null float64
ch-emp       3013 non-null float64
issue        3013 non-null float64
bm           3013 non-null float64
new-comp     3013 non-null int64
```

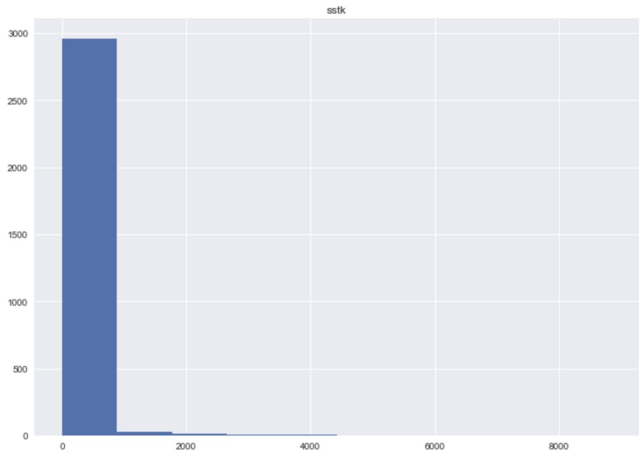


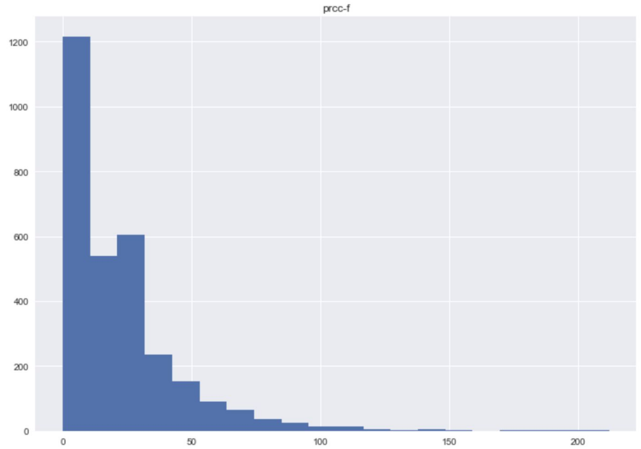
```
Not fraud (0),  2392
Fraud (1),  621
```

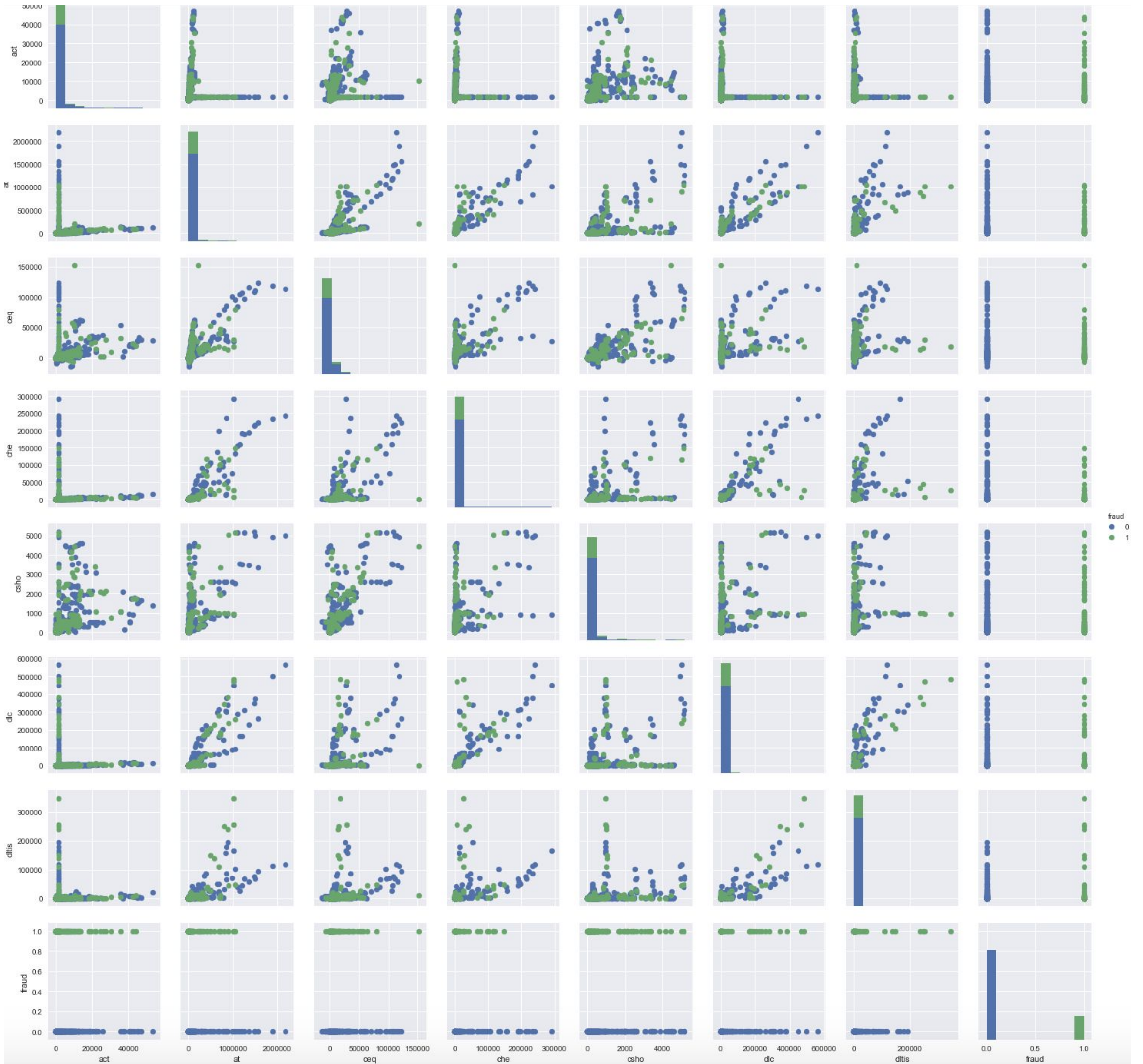


(corr >= 0.5) | (corr <= -0.4)

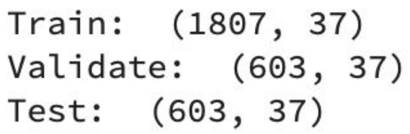



sstk
3000
2500
2000
1500
1000
500
0
0
2000
4000
6000
8000


prcc-f
1200
1000
800
600
400
200
0
0
50
100
150
200

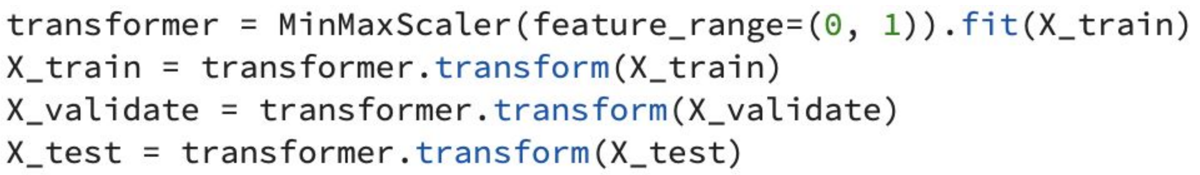

act
50000
40000
30000
20000
10000
0
at
2000000
1500000
1000000
500000
0
ceq
150000
100000
50000
0
che
300000
250000
200000
150000
100000
50000
0
csho
5000
4000
3000
2000
1000
0
dlc
600000
500000
400000
300000
200000
100000
0
dltis
300000
200000
100000
0
fraud
1.0
0.8
0.6
0.4
0.2
0.0
0
20000
40000
act
0
1000000
2000000
at
0
50000
100000
150000
ceq
0
100000
200000
300000
che
0
2000
4000
csho
0
200000
400000
600000
dlc
0
200000
dltis
0.0
0.5
1.0
fraud
fraud
0
1

1.



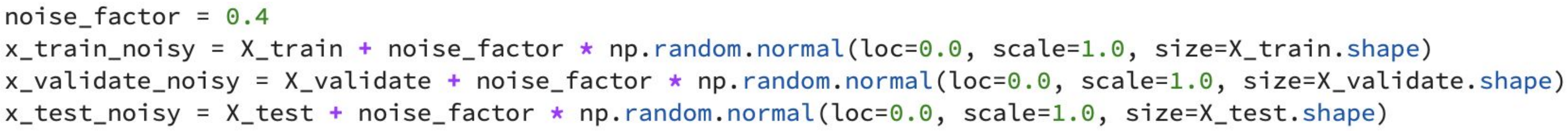
```
Train:  (1807, 37)
Validate:  (603, 37)
Test:  (603, 37)
```

2.



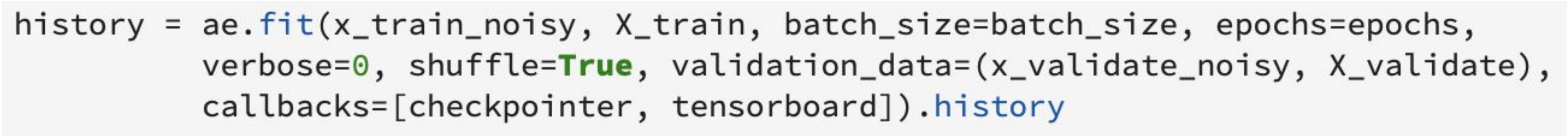
```
transformer = MinMaxScaler(feature_range=(0, 1)).fit(X_train)
X_train = transformer.transform(X_train)
X_validate = transformer.transform(X_validate)
X_test = transformer.transform(X_test)
```

3.



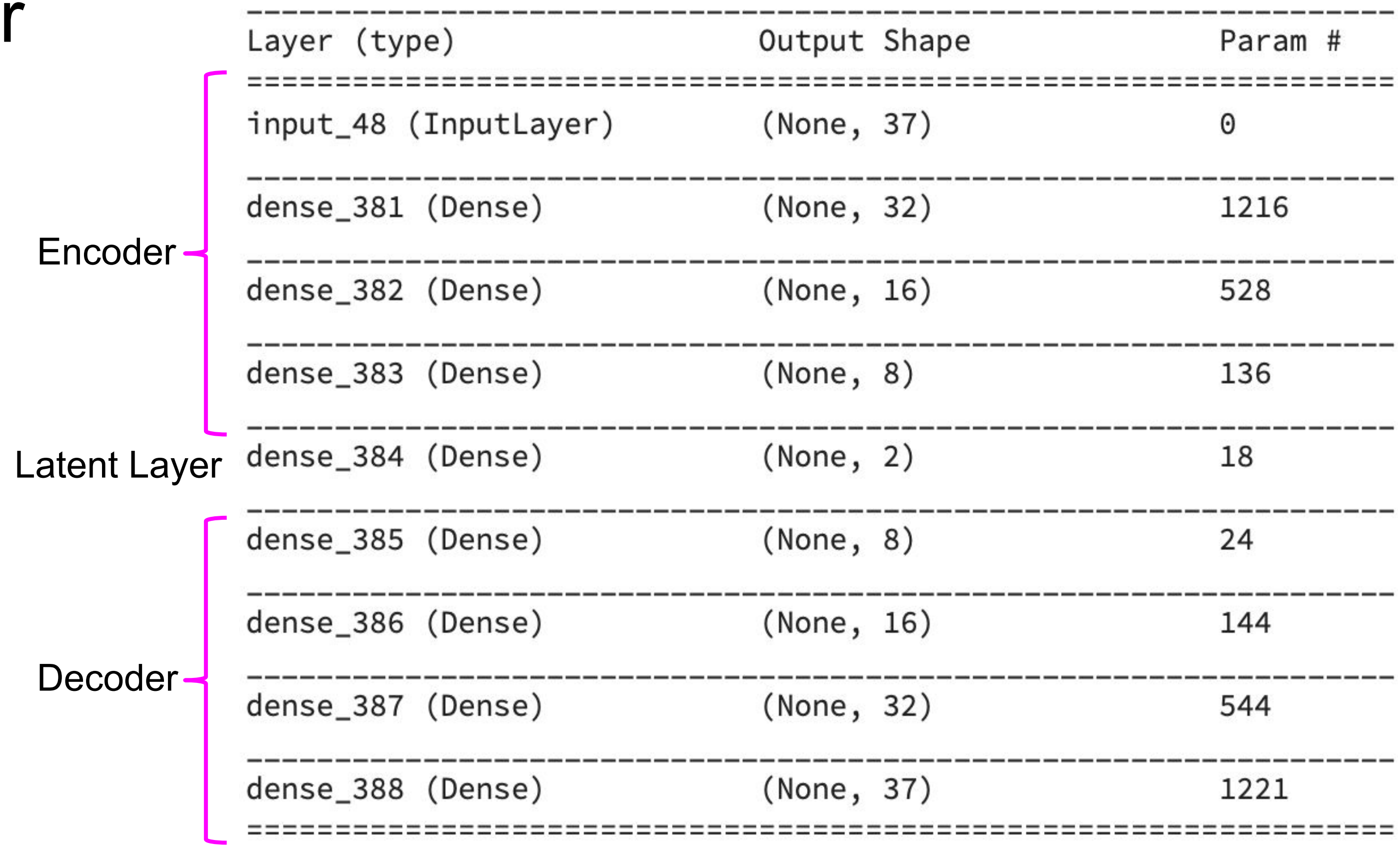
```
noise_factor = 0.4
x_train_noisy = X_train + noise_factor * np.random.normal(loc=0.0, scale=1.0, size=X_train.shape)
x_validate_noisy = X_validate + noise_factor * np.random.normal(loc=0.0, scale=1.0, size=X_validate.shape)
x_test_noisy = X_test + noise_factor * np.random.normal(loc=0.0, scale=1.0, size=X_test.shape)
```

4.



```
history = ae.fit(x_train_noisy, X_train, batch_size=batch_size, epochs=epochs,
                 verbose=0, shuffle=True, validation_data=(x_validate_noisy, X_validate),
                 callbacks=[checkpointer, tensorboard]).history
```

Autoencoder



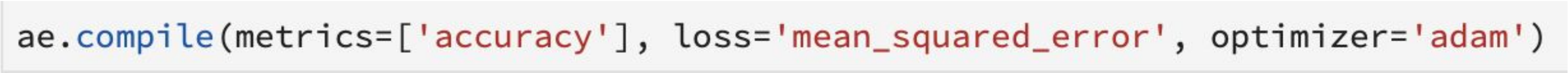
```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_48 (InputLayer)        (None, 37)                0
_________________________________________________________________
dense_381 (Dense)            (None, 32)                1216
_________________________________________________________________
dense_382 (Dense)            (None, 16)                528
_________________________________________________________________
dense_383 (Dense)            (None, 8)                 136
_________________________________________________________________
dense_384 (Dense)            (None, 2)                 18
_________________________________________________________________
dense_385 (Dense)            (None, 8)                 24
_________________________________________________________________
dense_386 (Dense)            (None, 16)                144
_________________________________________________________________
dense_387 (Dense)            (None, 32)                544
_________________________________________________________________
dense_388 (Dense)            (None, 37)                1221
=================================================================
```

Encoder

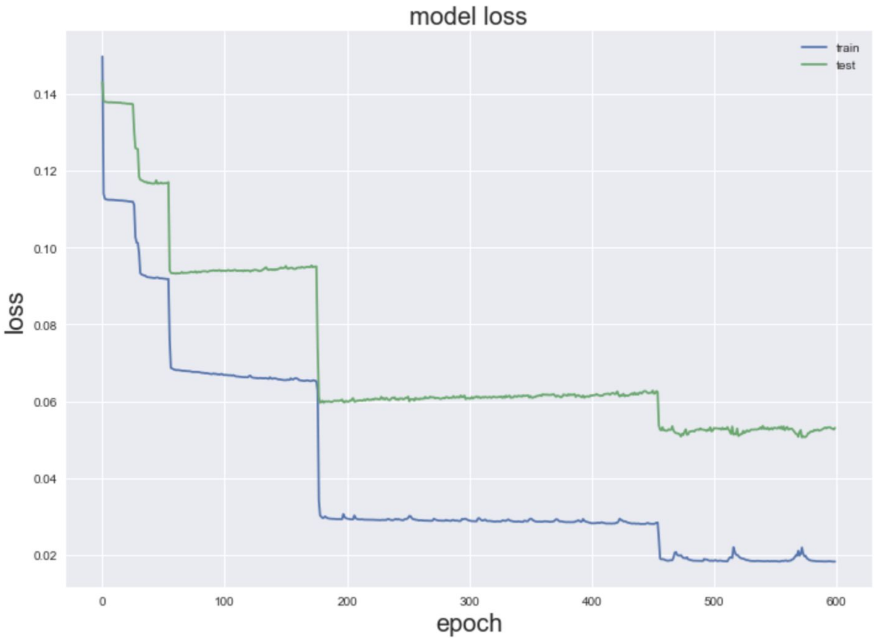
Latent Layer

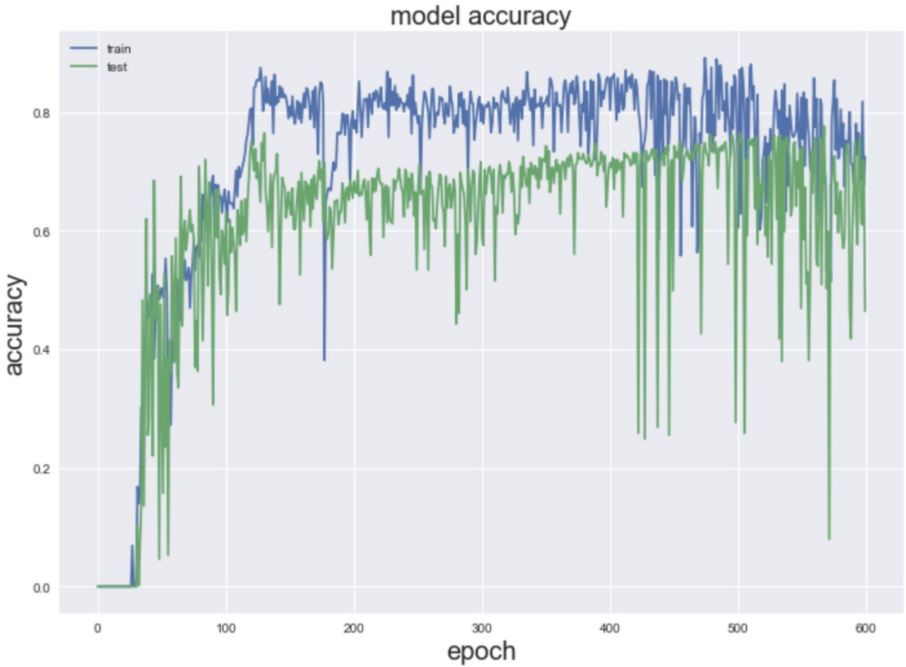
Decoder

Train & Validation

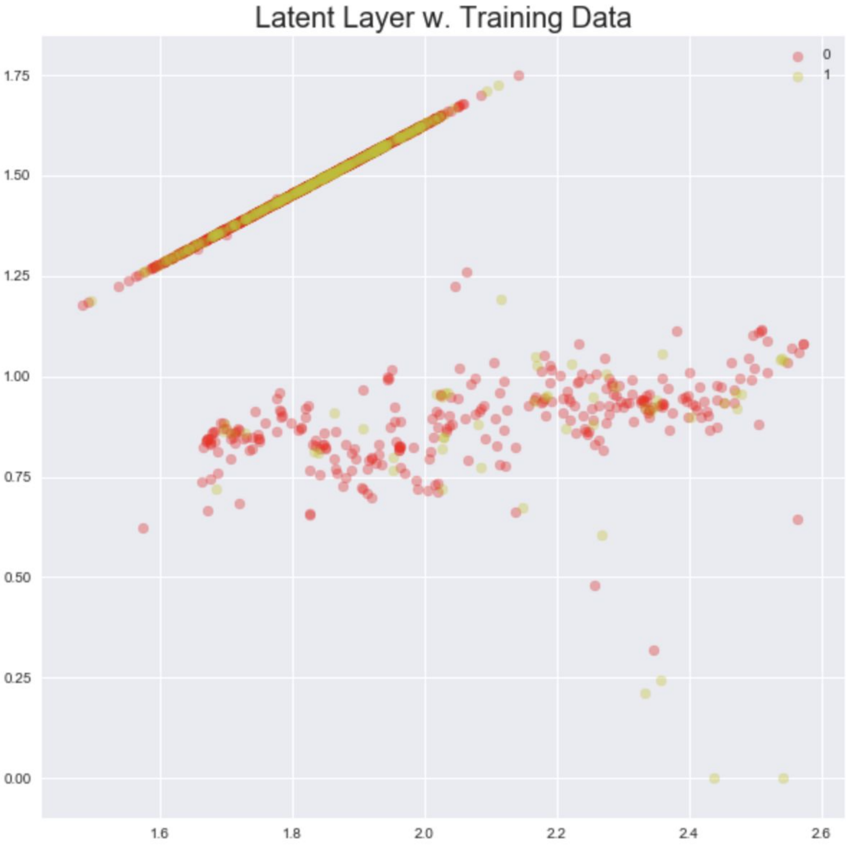


```
ae.compile(metrics=['accuracy'], loss='mean_squared_error', optimizer='adam')
```





Clustering with Denoising Autoencoders


Latent Layer w. Training Data