Analysis

January 5, 2022

1 Analysis for California Dataset

[6]:

```
import warnings
warnings.filterwarnings('ignore')
```

[7]:

```
import scrapbook as sb
import pandas as pd
import numpy as np
import seaborn as sns
import numpy as np
from statistics import mean
import matplotlib.pyplot as plt
```

1.1 BaseLine Models

We have used Random Forest, Catboost, Vanilla NN and Stats Model as the Baseline Model for the problem

Importing data of baseline Models

[21]:

```
books = sb.read_notebooks("./BaseLine_Model_Output")
baseLine_data = []
for nb in books.notebooks:
    nbList=[nb.scraps['Catboost MSE'].data]
    baseLine_data.append(nbList)
df = pd.DataFrame(baseLine_data, columns = ["Catboost"])
display(df)
print("MEAN:")
print(df.mean(axis = 0))
baseLine_data = np.array(baseLine_data)
```

```
   Catboost
0  0.140407
1  0.155261
2  0.151374
3  0.156897
4  0.146647
5  0.146694
6  0.137142
```

```
7   0.149315
8   0.142645
9   0.143583
```

```
MEAN:
Catboost    0.146996
dtype: float64
```

1.2 GAN

Simple C-GAN was used to train the dataset

[11]:

```python
book = sb.read_notebooks("./GAN_Output")
gan_data = []
gan_mse = []
for nb in book.notebooks:
    metrics = nb.scraps['GAN_1 Metrics'].data
    for i in range(1000):
        gan_mse.append(metrics[0][i])
    nbList = [nb.scraps['GAN Model MSE'].data,
              nb.scraps['GAN Model MAE'].data,
              nb.scraps['GAN Model Euclidean distance'].data,
              nb.scraps['GAN Model Manhattan Distance'].data]
    gan_data.append(nbList)
df = pd.DataFrame(gan_data, columns = ['MSE','MAE','Euclidean␣
 ↪Distance','Manhattan Distance'])
display(df)
print("MEAN:")
print(df.mean(axis = 0))
gan_data = np.array(gan_data)
```

```
        MSE       MAE  Euclidean Distance  Manhattan Distance
0  0.344669  0.382632           37.718008         1579.503514
1  0.315276  0.386188           36.074626         1594.183435
2  0.388851  0.415491           40.062468         1715.148223
3  0.461058  0.471094           43.624342         1944.677065
4  0.334432  0.395961           37.153845         1634.526491
5  0.383356  0.398691           39.778135         1645.796486
6  0.463805  0.465104           43.752859         1919.947921
7  0.348313  0.408673           37.917110         1687.001492
8  0.350196  0.396515           38.019089         1636.812341
9  0.386557  0.410137           39.944970         1693.043660
```

```
MEAN:
MSE                      0.377651
MAE                      0.413048
Euclidean Distance      39.404545
Manhattan Distance    1705.064063
dtype: float64
```

1.3 ABC_GAN (Catboost Pre generator)

[19]:

```
books = sb.read_notebooks("./ABC_GAN_Output")
books_skip = sb.read_notebooks("./ABC_GAN_Skip_Output")
paramVal = [0.01,0.1,1]

#Simple ABC GAN
abc_mse = [[] for i in range(3)]
abc_mse_mean = [[] for i in range(3)]

for nb in books.notebooks:
    metrics1 = np.array(nb.scraps['ABC_GAN_1 Metrics'].data)
    paramVar = float(nb.papermill_dataframe.iloc[0]['value'])
    for i in range(3):
        if paramVar == paramVal[i]:
            for j in range(100):
                abc_mse[i].append(metrics1[0,j])
            abc_mse_mean[i].append(mean(metrics1[0,:]))

#ABC GAN with skip connection
abc_mse_skip_mean = [[] for i in range(3)]
abc_mse_skip = [[] for i in range(3)]
abc_weights = [[] for i in range(3)]

for nb in books_skip.notebooks:
    metrics3 = np.array(nb.scraps['ABC_GAN_3 Metrics'].data)
    paramVar = float(nb.papermill_dataframe.iloc[0]['value'])
    #Divide data according to parameters
    for i in range(3):
        if paramVar == paramVal[i]:
            for j in range(100):
                abc_mse_skip[i].append(metrics3[0,j])
            abc_weights[i].append(nb.scraps['Skip Connection Weight'].data)
            abc_mse_skip_mean[i].append(mean(metrics3[0,:]))
```

[20]:

```
for i in range(3):
    data = []
    var = var = paramVal[i]
    for j in range(len(abc_weights[i])):
        data.
 ↪append([abc_mse_mean[i][j],abc_mse_skip_mean[i][j],abc_weights[i][j]])
    print("y_gan = y_abc + N(0,"+str(var)+")")
    df = pd.DataFrame(data, columns = ['ABC_GAN','ABC_GAN(Skip␣
 ↪Connection)','Weight(Skip Connection)'])
    display(df)
    print(df.mean(axis=0))
```

```
y_gan = y_abc + N(0,0.01)
```

```
        ABC_GAN  ABC_GAN(Skip Connection)  Weight(Skip Connection)
0  1.486061e+08              6.157607e+06                 0.103987
1  1.687264e+00              2.476996e+04                 0.008723
2  2.257249e+09              8.619717e-02                 0.000000
3  2.736604e+07              9.428696e-02                 0.000000
4  1.504063e+06              8.242447e-02                 0.010924
5  8.086260e+03              1.192063e+07                 0.932922
6  1.645717e+00              9.518305e-02                 0.000000
7  1.154148e+00              1.242531e+12                 0.625253
8  1.032915e+00              9.441970e-02                 0.000000
9  1.273397e+00              1.880617e+09                 0.387284

ABC_GAN                     2.434733e+08
ABC_GAN(Skip Connection)    1.244429e+11
Weight(Skip Connection)     2.069092e-01
dtype: float64
y_gan = y_abc + N(0,0.1)

        ABC_GAN  ABC_GAN(Skip Connection)  Weight(Skip Connection)
0  1.029524e+00              1.003677e-01                 0.000000
1  1.182904e+00              9.854411e-02                 0.000000
2  7.136822e+06              1.026498e-01                 0.000000
3  9.928643e-01              1.067837e-01                 0.000000
4  9.729196e+05              1.780603e+06                 0.081307
5  6.960694e+00              1.026660e-01                 0.000000
6  1.573508e+00              1.051708e-01                 0.000000
7  3.144130e-01              3.391387e+07                 0.815565
8  4.810671e+04              9.883938e-02                 0.000000

ABC_GAN                     9.064290e+05
ABC_GAN(Skip Connection)    3.966053e+06
Weight(Skip Connection)     9.965248e-02
dtype: float64
y_gan = y_abc + N(0,1)

         ABC_GAN  ABC_GAN(Skip Connection)  Weight(Skip Connection)
0       0.514911              4.484122e-01                 0.386985
1       1.115848              5.705249e-01                 0.502244
2       1.612658              3.487498e+08                 0.558135
3       1.092401              5.799999e-01                 0.535517
4  141248.397180              6.862736e+08                 0.600012
5       8.665157              6.001428e-01                 0.497984
6       2.195859              1.086587e+00                 0.000000
7       2.204236              3.406594e-01                 0.086324
8       0.381689              5.937615e-01                 0.454289
9       8.426801              1.086495e+00                 0.000000

ABC_GAN                     1.412746e+04
ABC_GAN(Skip Connection)    1.035023e+08
Weight(Skip Connection)     3.621489e-01
```

```
dtype: float64
```

[ ]: