Assignment_10

November 9, 2020

```
[5]: # Method 1 DD and PD Descriptive Stats

     kmv1_agg_dd = pd.read_pickle('kmv1_agg_dd.pkl')
     kmv1_agg_pd = pd.read_pickle('kmv1_agg_pd.pkl')
     kmv1_agg_dd
     kmv1_agg_pd
```

```
[5]:           dd_naive1       dd_naive2       dd_naive3
     count  192031.000000   192031.000000   192031.000000
     mean        5.869934        5.604568        6.127418
     std         5.866646        5.804907        9.115865
     min        -9.010434       -9.125004       -9.009234
     q25         2.248173        1.970016        2.391616
     median      4.851107        4.510630        5.116926
     q75         8.181489        7.848818        8.545329
     max       782.608265      781.009107     2985.099231
```

```
[5]:           pd_naive1       pd_naive2       pd_naive3
     count   1.920310e+05    1.920310e+05    1.920310e+05
     mean    7.849295e-02    8.623026e-02    7.615972e-02
     std     2.008657e-01    2.032773e-01    2.014200e-01
     min     0.000000e+00    0.000000e+00    0.000000e+00
     q25     1.401788e-16    2.099884e-15    6.408546e-18
     median  6.138719e-07    3.231771e-06    1.552781e-07
     q75     1.228259e-02    2.441826e-02    8.387197e-03
     max     1.000000e+00    1.000000e+00    1.000000e+00
```

```
[6]: # Method 2 DD and PD Descriptive Stats

     kmv2_agg = pd.read_pickle('kmv2_agg.pkl')
     kmv2_agg
```

```
[6]:              dd_bs           pd_bs
     count  180245.000000    1.802450e+05
     mean        6.982726    5.441718e-02
     std        32.199882    1.835348e-01
     min     -3043.085480    0.000000e+00
     q25         3.154158    4.249452e-21
```

```
median        5.879465  2.057979e-09
q75           9.353257  8.048101e-04
max        9097.011641  1.000000e+00
```

[9]: ```python
Method 1 and 2 DD Correlations

corr_12_dd = pd.read_pickle('corr_12_dd.pkl')
corr_12_dd
```

[9]:
|          | dd_naive1 | dd_naive2 | dd_naive3 | dd_bs    |
|----------|-----------|-----------|-----------|----------|
| dd_naive1 | 1.000000  | 0.998194  | 0.648218  | 0.194526 |
| dd_naive2 | 0.998194  | 1.000000  | 0.644652  | 0.187849 |
| dd_naive3 | 0.648218  | 0.644652  | 1.000000  | 0.272064 |
| dd_bs    | 0.194526  | 0.187849  | 0.272064  | 1.000000 |

[10]: ```python
# Method 1 and 2 PD Correlations

corr_12_pd = pd.read_pickle('corr_12_pd.pkl')
corr_12_pd
```

[10]:
	pd_naive1	pd_naive2	pd_naive3	pd_bs
pd_naive1	1.000000	0.992904	0.998445	0.980827
pd_naive2	0.992904	1.000000	0.985702	0.961190
pd_naive3	0.998445	0.985702	1.000000	0.985183
pd_bs	0.980827	0.961190	0.985183	1.000000

The first and second naive distance to default calculations are highly correlated. The third naive distance to default calculation is very weakly correlated with the black scholes direct solving method. For the probability to default, all the method values are highly correlated. The black scholes direct solver is most closely correlated to the third naive method.
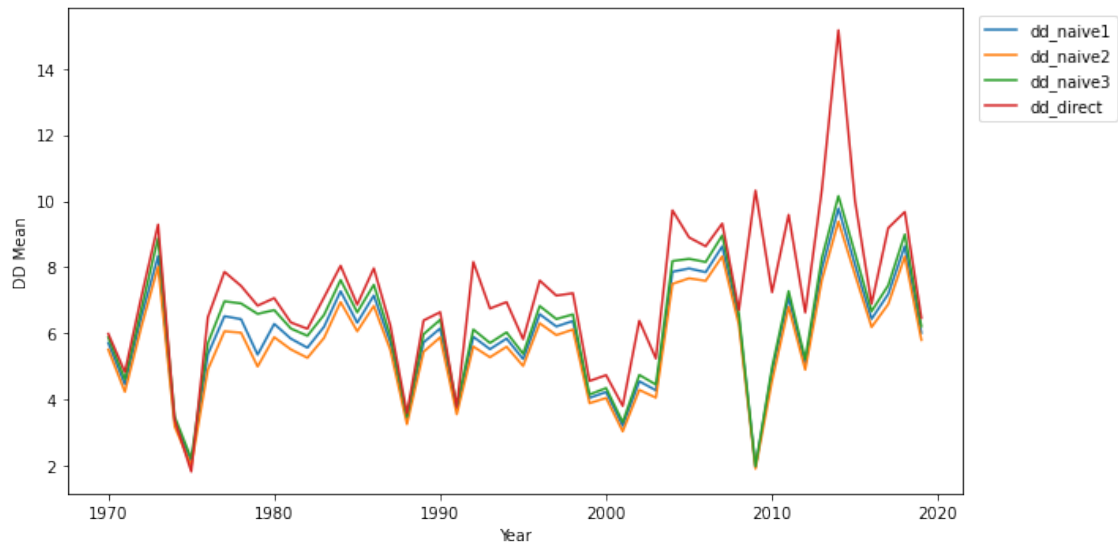
[14]: ```python
DD Mean for each Year for Method 1 and Method 2

plot_df(df_12_list[0], True, 4, 'DD Mean', None, None, None, fig_size=(10, 5),
        legend_loc='upper left', legend_bbox=(1.01, 1))
```

```
<Figure size 432x288 with 0 Axes>
```

```
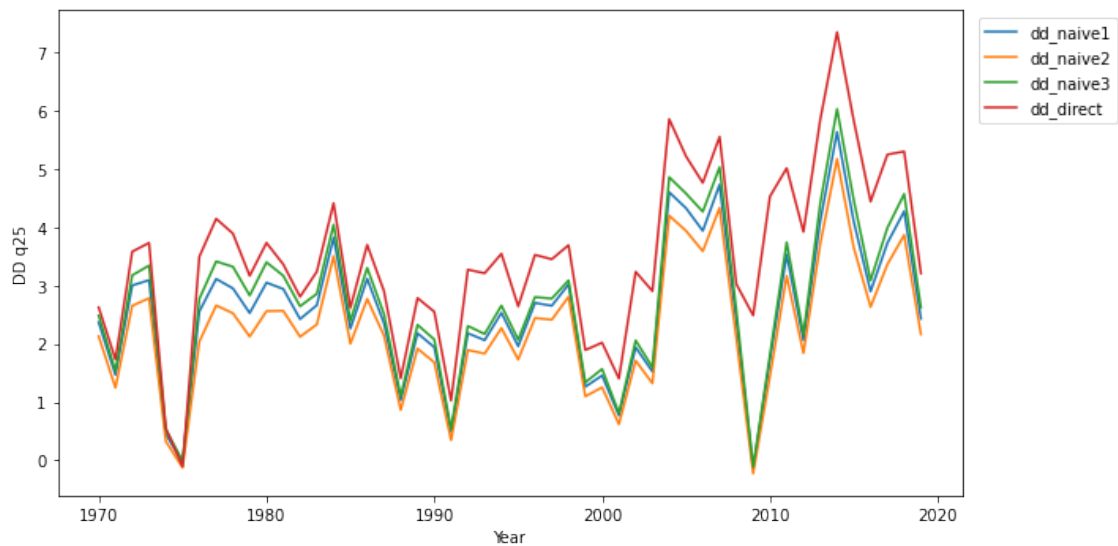[15]: # DD q25 for each Year for Method 1 and Method 2

      plot_df(df_12_list[1], True, 4, 'DD q25', None, None, None, fig_size=(10, 5),
              legend_loc='upper left', legend_bbox=(1.01, 1))
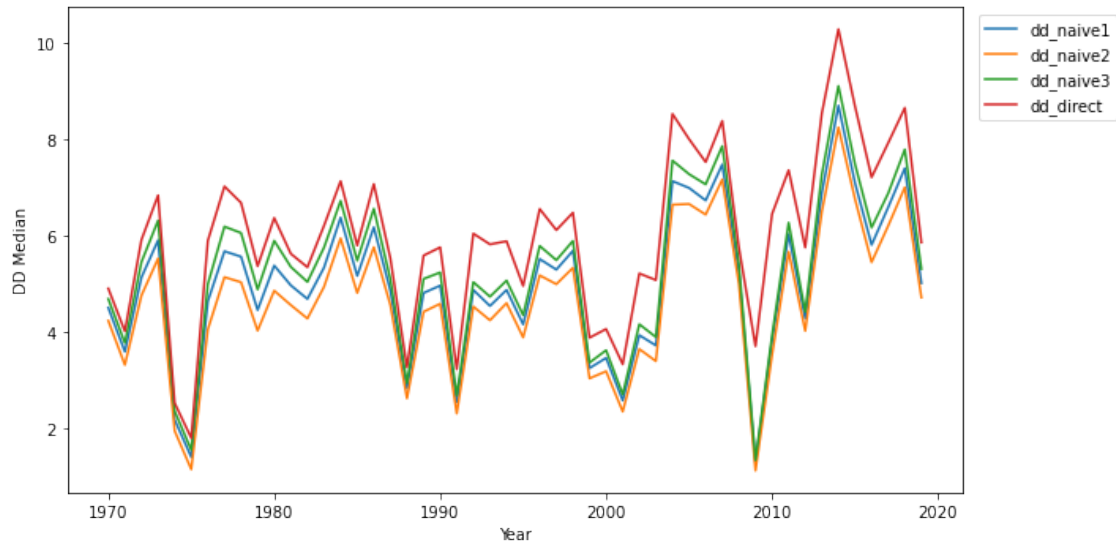```

<Figure size 432x288 with 0 Axes>



```
[16]: # DD Median for each Year for Method 1 and Method 2

      plot_df(df_12_list[2], True, 4, 'DD Median', None, None, None, fig_size=(10, 5),
```

```
          legend_loc='upper left', legend_bbox=(1.01, 1))
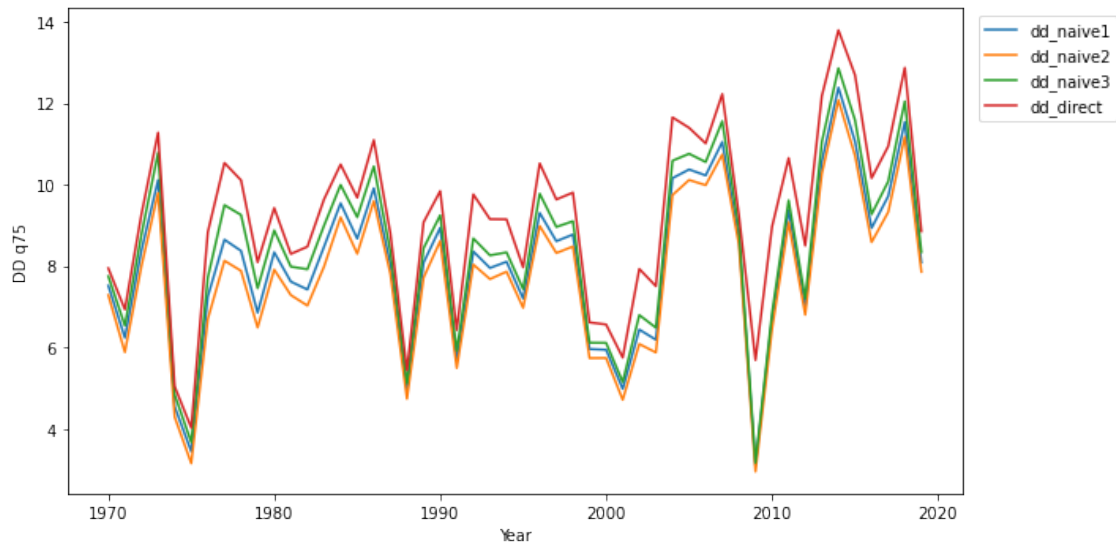```

<Figure size 432x288 with 0 Axes>



[17]:
```
DD q75 for each Year for Method 1 and Method 2

plot_df(df_12_list[3], True, 4, 'DD q75', None, None, None, fig_size=(10, 5),
        legend_loc='upper left', legend_bbox=(1.01, 1))
```

<Figure size 432x288 with 0 Axes>

The direct solving method distance to default values are almost always higher than the naive methods. Out of the naive methods, naive method 3 is usually the largest distance to default, naive method 1 is in the middle, and naive method 2 is the smallest distance to default. When it comes to the mean values, around year 2007 or 2008, there is a large spike in the mean for the direct solving method, while the naive values all show a dip in the mean. This dip in the mean in 2007 or 2008 also occurs in the q25, median, and q75 graphs. It looks like there are some outlier distance to default values in the direct solving method influencing the mean heavily.

## 0.1 Plots and Tables with NBER Recession/BAAFFM/CFSI data are at the end

```
[19]:  # Method 3 DD and PD Descriptive Stats


       kmv3_agg = pd.read_pickle('kmv3_agg.pkl')
       kmv3_agg
```

```
[19]:            dd_iter         pd_iter
       count   9855.000000   9.855000e+03
       mean       6.751938   8.213964e-02
       std        6.274880   2.222606e-01
       min     -143.307661   0.000000e+00
       q25        2.637360   9.267429e-23
       median     5.821136   2.922442e-09
       q75        9.749519   4.177707e-03
       max       59.382654   1.000000e+00
```

```
[20]:  # Method 1, 2, and 3 DD Correlations


       corr_123_dd = pd.read_pickle('corr_123_dd.pkl')
       corr_123_dd
```

```
[20]:            dd_naive1  dd_naive2  dd_naive3     dd_bs    dd_iter
       dd_naive1   1.000000   0.998705   0.997493  0.180156   0.903756
       dd_naive2   0.998705   1.000000   0.993067  0.177675   0.892648
       dd_naive3   0.997493   0.993067   1.000000  0.182637   0.914417
       dd_bs       0.180156   0.177675   0.182637  1.000000   0.192715
       dd_iter     0.903756   0.892648   0.914417  0.192715   1.000000
```

```
[21]:  # Method 1, 2, and 3 PD Correlations


       corr_123_pd = pd.read_pickle('corr_123_pd.pkl')
       corr_123_pd
```

```
[21]:            pd_naive1  pd_naive2  pd_naive3     pd_bs    pd_iter
       pd_naive1   1.000000   0.993532   0.998726  0.982514   0.957052
       pd_naive2   0.993532   1.000000   0.987355  0.964107   0.943974
       pd_naive3   0.998726   0.987355   1.000000  0.986809   0.959277
       pd_bs       0.982514   0.964107   0.986809  1.000000   0.957148
```

```
pd_iter     0.957052   0.943974   0.959277   0.957148   1.000000
```

For the firm-years in 1990-2020 and the random sampled 1000 firms, there are some differing trends than the earlier correlations. There already weak correlation between the black scholes direct solving method is even weaker. Again, this could be due to a some significant outliers in the direct solving method. The iterative method is most closely correlated with the 3rd naive method. The iterative probability of default is very strongly correlated amongst all the other probability to default values.

[23]:
```python
# DD Mean for each Year for Method 1, Method 2, and Method 3

plot_df(df_123_list[0], True, 5, 'DD Mean', None, None, None, fig_size=(10, 5),
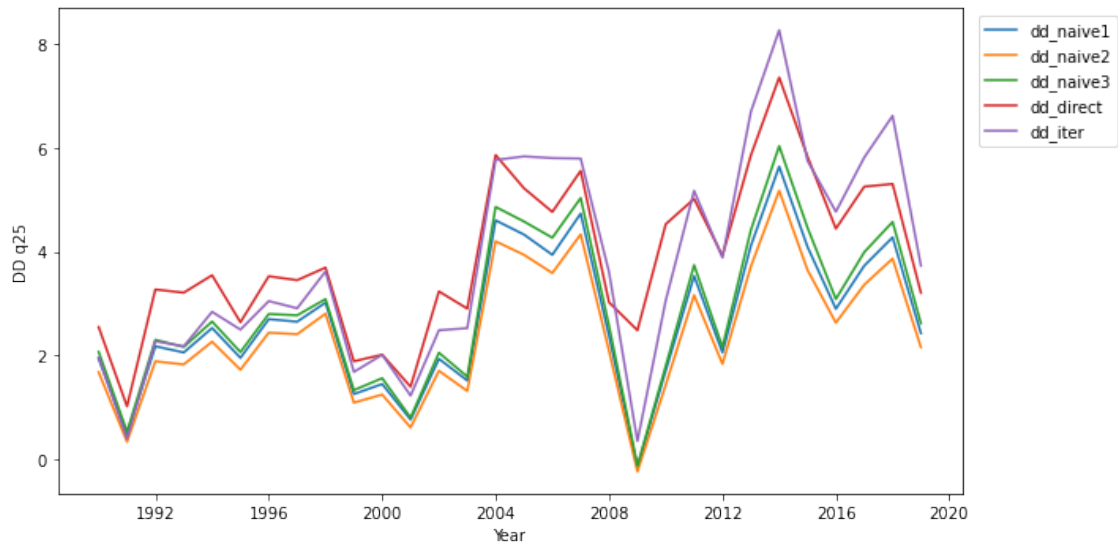        legend_loc='upper left', legend_bbox=(1.01, 1))
```

```
<Figure size 432x288 with 0 Axes>
```
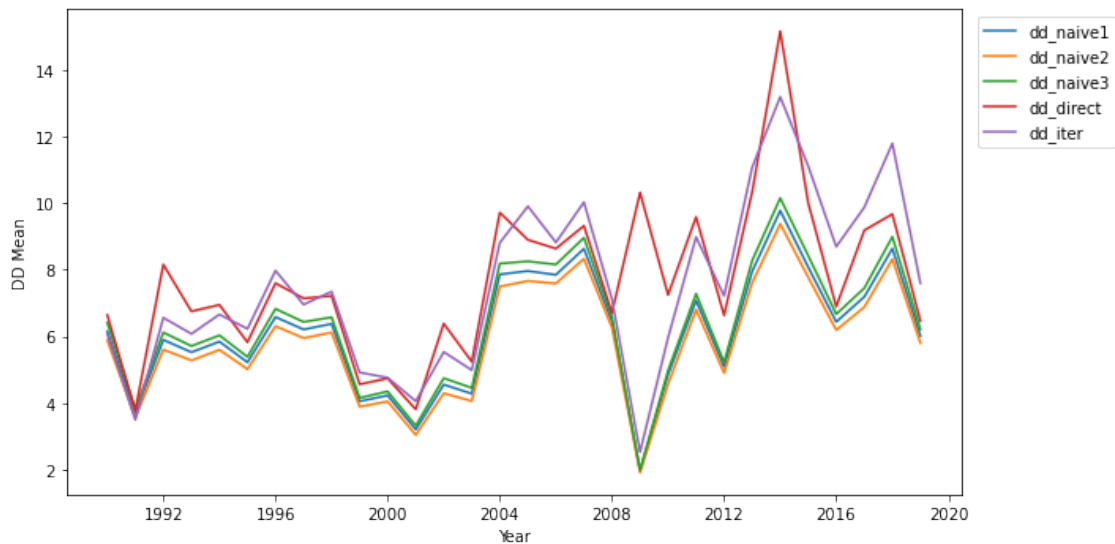

[24]:
```python
# DD q25 for each Year for Method 1, Method 2, and Method 3

plot_df(df_123_list[1], True, 5, 'DD q25', None, None, None, fig_size=(10, 5),
        legend_loc='upper left', legend_bbox=(1.01, 1))
```

```
<Figure size 432x288 with 0 Axes>
```

[25]:
```
# DD Median for each Year for Method 1, Method 2, and Method 3

plot_df(df_123_list[2], True, 5, 'DD Median', None, None, None, fig_size=(10,␣
 ↪5),
        legend_loc='upper left', legend_bbox=(1.01, 1))
```

<Figure size 432x288 with 0 Axes>

[26]:
```
# DD q75 for each Year for Method 1, Method 2, and Method 3
```

```
plot_df(df_123_list[3], True, 5, 'DD q75', None, None, None, fig_size=(10, 5),
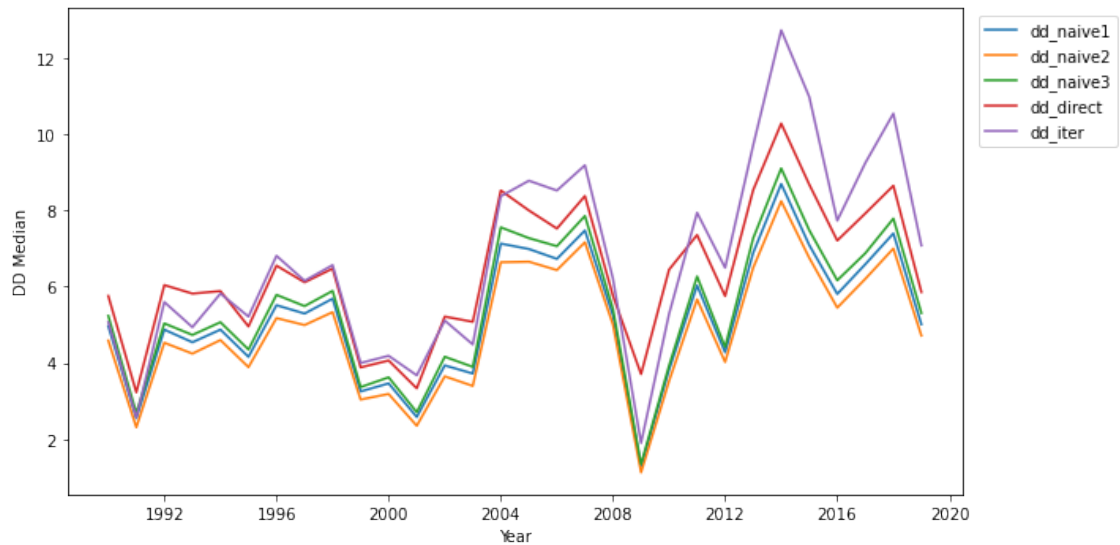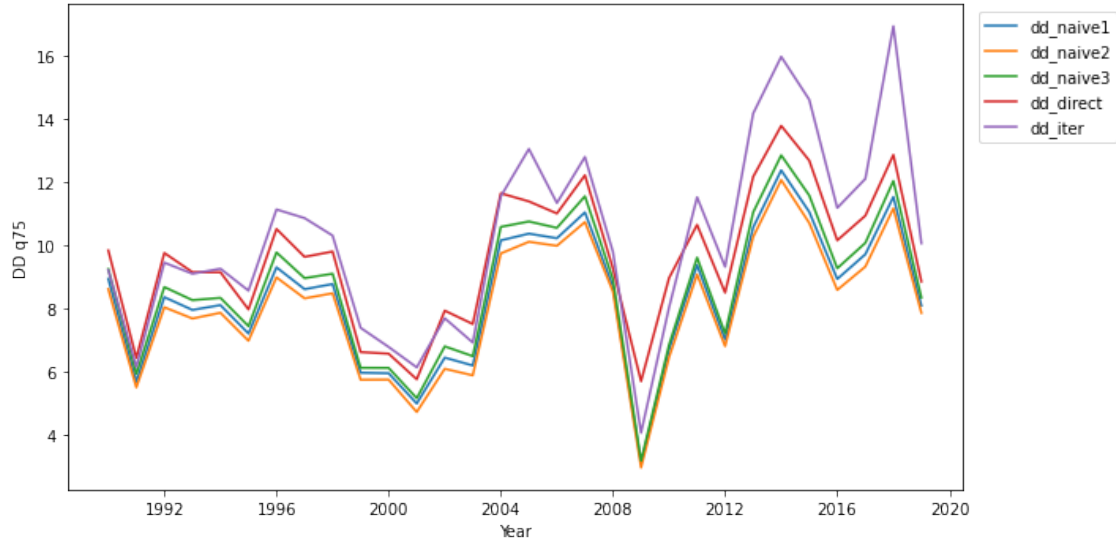        legend_loc='upper left', legend_bbox=(1.01, 1))
```

<Figure size 432x288 with 0 Axes>

The iterative method distance to default calculations are generally similar to the direct solving method values. It does not appear to be more consistently larger or smaller than the direct solving. It is almost always larger than all the naive method distance to default values though.

[27]:
```
# Method 1 DD and PD Descriptive Stats for NBER Recession = 1

kmv1_rec1_dd
kmv1_rec1_pd
```

[27]:
	dd_naive1	dd_naive2	dd_naive3
count	24067.000000	24067.000000	24067.000000
mean	4.499108	4.291992	4.694478
std	7.131849	7.092051	7.486124
min	-5.095763	-3.295008	-7.705432
q25	1.241035	1.038200	1.340101
median	3.445964	3.150472	3.669276
q75	6.406428	6.087949	6.735391
max	782.608265	781.009107	843.751720

[27]:
	pd_naive1	pd_naive2	pd_naive3
count	2.406700e+04	2.406700e+04	2.406700e+04
mean	1.339617e-01	1.421649e-01	1.310588e-01
std	2.603314e-01	2.578472e-01	2.622161e-01
min	0.000000e+00	0.000000e+00	0.000000e+00

8

```
q25      7.448708e-11  5.718330e-10  8.174516e-12
median   2.845128e-04  8.150353e-04  1.216190e-04
q75      1.072964e-01  1.495884e-01  9.010632e-02
max      9.999998e-01  9.995079e-01  1.000000e+00
```

`[28]:` *# Method 1 DD and PD Descriptive Stats for NBER Recession = 0*

```
kmv1_rec0_dd
kmv1_rec0_pd
```

`[28]:`

	dd_naive1	dd_naive2	dd_naive3
count	167964.000000	167964.000000	167964.000000
mean	6.066355	5.792643	6.332739
std	5.635027	5.571033	9.308054
min	-9.010434	-9.125004	-9.009234
q25	2.429989	2.140018	2.582968
median	5.052460	4.709231	5.330385
q75	8.395996	8.071891	8.776647
max	303.571116	303.564969	2985.099231

`[28]:`

	pd_naive1	pd_naive2	pd_naive3
count	1.679640e+05	1.679640e+05	1.679640e+05
mean	7.054502e-02	7.821557e-02	6.829342e-02
std	1.895091e-01	1.928831e-01	1.898363e-01
min	0.000000e+00	0.000000e+00	0.000000e+00
q25	2.309801e-17	3.460889e-16	8.421053e-19
median	2.180774e-07	1.243264e-06	4.900227e-08
q75	7.549631e-03	1.617665e-02	4.897717e-03
max	1.000000e+00	1.000000e+00	1.000000e+00

`[29]:` *# Method 2 DD and PD Descriptive Stats for NBER Recession = 1*

```
kmv2_rec1
```

`[29]:`

	dd_bs	pd_bs
count	21693.000000	2.169300e+04
mean	5.809968	9.240327e-02
std	44.843525	2.383048e-01
min	-409.733278	0.000000e+00
q25	2.153647	2.037441e-14
median	4.502837	3.352611e-06
q75	7.558582	1.563393e-02
max	5998.446174	1.000000e+00

`[30]:` *# Method 2 DD and PD Descriptive Stats for NBER Recession = 0*

```
kmv2_rec0
```

```
[30]:              dd_bs          pd_bs
      count  158552.000000  1.585520e+05
      mean        7.143182  4.921995e-02
      std        30.055834  1.740685e-01
      min     -3043.085480  0.000000e+00
      q25         3.320130  5.409687e-22
      median      6.084459  5.844259e-10
      q75         9.568762  4.498775e-04
      max      9097.011641  1.000000e+00
```

```
[31]:  # Method 3 DD and PD Descriptive Stats for NBER Recession = 1

       kmv3_rec1
```

```
[31]:            dd_iter       pd_iter
      count  724.000000  7.240000e+02
      mean     4.497990  1.478861e-01
      std      4.807508  2.910812e-01
      min    -15.593645  2.376437e-279
      q25      1.309890  2.137755e-11
      median   3.779055  7.873838e-05
      q75      6.594222  9.511664e-02
      max     35.694843  1.000000e+00
```

```
[32]:  # Method 3 DD and PD Descriptive Stats for NBER Recession = 0

       kmv3_rec0
```

```
[32]:             dd_iter       pd_iter
      count  9131.000000  9.131000e+03
      mean      6.930655  7.692658e-02
      std       6.342823  2.150293e-01
      min    -143.307661  0.000000e+00
      q25       2.779045  1.013166e-23
      median    6.024261  8.494210e-10
      q75       9.971747  2.725950e-03
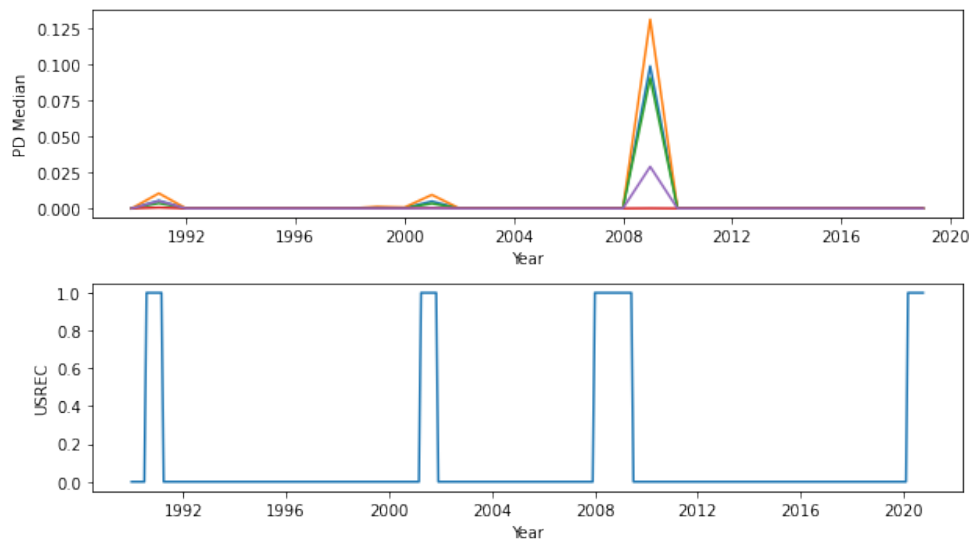      max      59.382654  1.000000e+00
```

```
[34]:  # DD Median values with NBER Recession Data
       plot_df(df_123_list[2], True, 5, 'DD Median', usrec_1990, None, None,␣
        ↪fig_size=(10, 5),
               legend_loc='upper left', legend_bbox=(1.01, 1))
```

```
<Figure size 432x288 with 0 Axes>
```
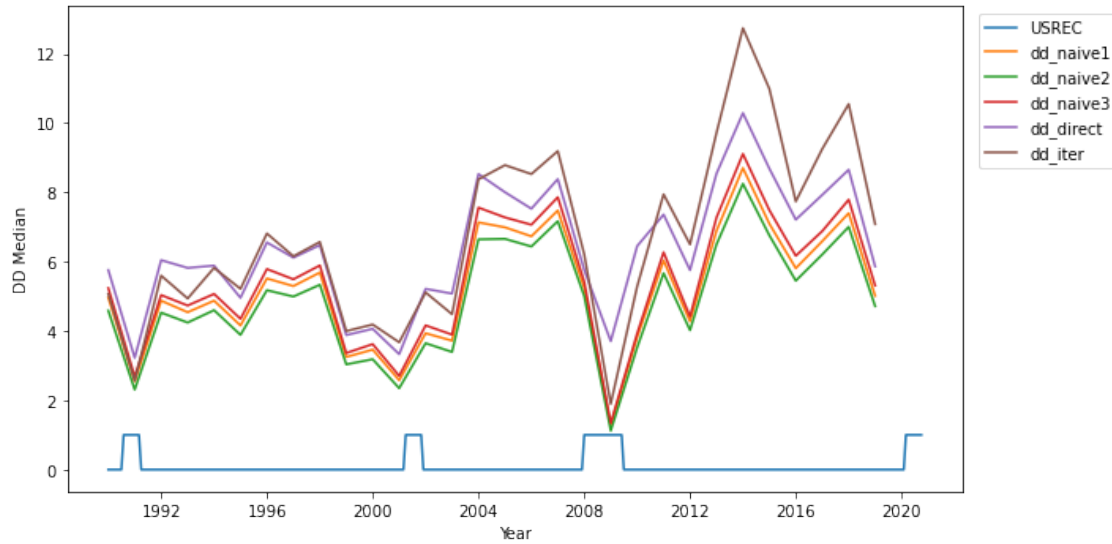
```
[35]:  # PD Median values with NBER Recession Data
       plot_df(df_123_median_pd, True, 5, 'PD Median', usrec_1990, None, None,␣
        ↪sub_plot=True,
               fig_size=(10, 5), legend_loc='upper left', legend_bbox=(1.01, 1))
```

<Figure size 432x288 with 0 Axes>



For all the distance to default calculation methods, when NBER Recession is 1, the distance to
default calculations are smaller and the probability to default values are higher than when the
NBER recession is 0. This occurs for the mean, q25, median, and q75.

Additionally in the graphs, we can see the distance to default median decreasing when the recession indicator is 1. The probability to default median also drastically increases during the recession inidicator being 1 in 1990-1991, 2001, and the great recession in 2008.

```python
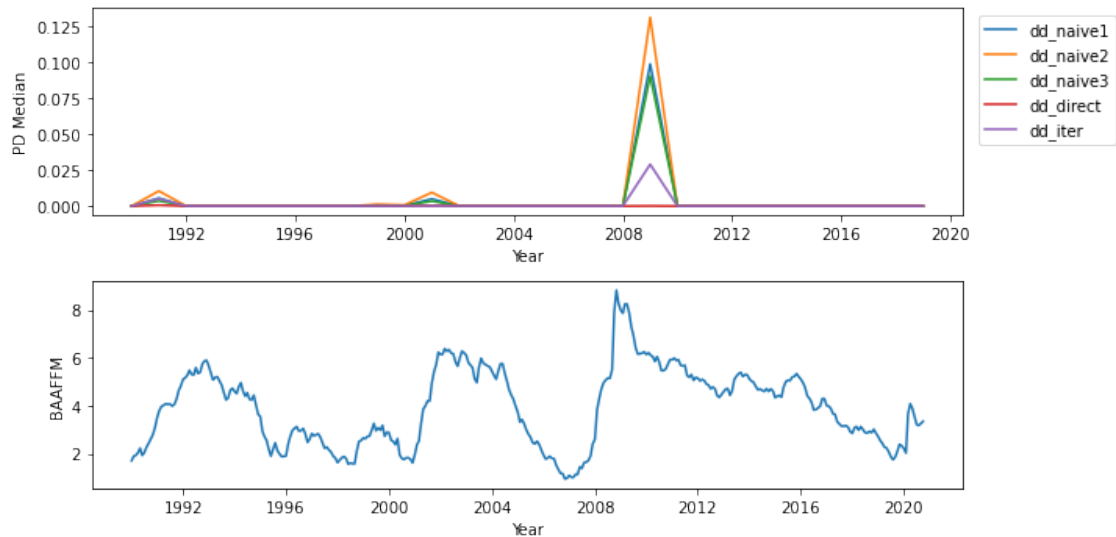[36]: # DD Median values with Moody's BAA-Fed Fund Spread
      plot_df(df_123_list[2], True, 5, 'DD Median', None, baaffm_1990, None,
       →fig_size=(10, 5),
              legend_loc='upper left', legend_bbox=(1.01, 1))
```

<Figure size 432x288 with 0 Axes>
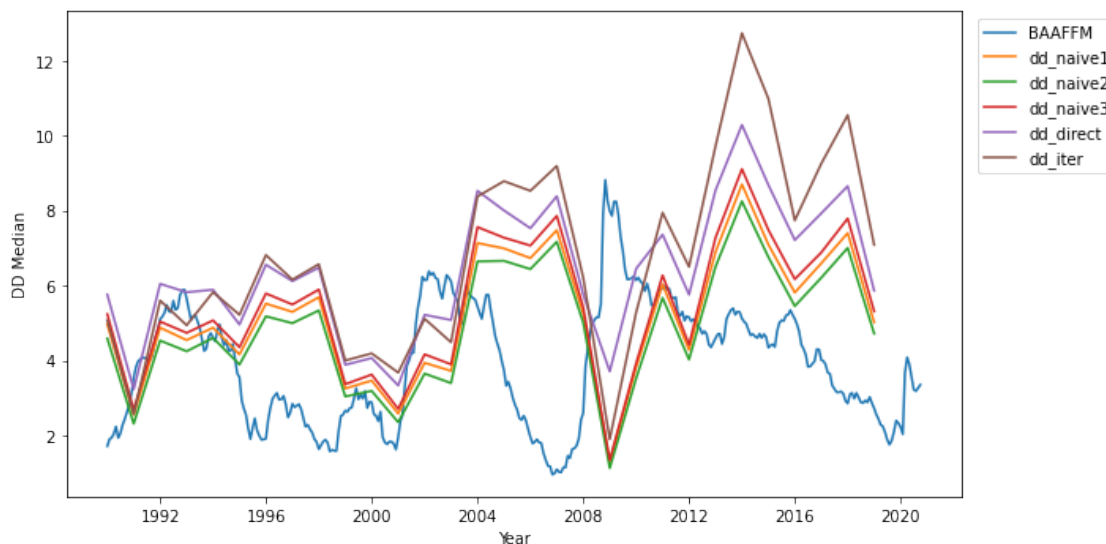

```python
[37]: # PD Median values with Moody's BAA-Fed Fund Spread
      plot_df(df_123_median_pd, True, 5, 'PD Median', None, baaffm_1990, None,
       →sub_plot=True,
              fig_size=(10, 5), legend_loc='upper left', legend_bbox=(1.01, 1))
```

<Figure size 432x288 with 0 Axes>

The distance to default values seem to occasionallly inversely mirror the Moody's BAA-Fed Fund Spread. We see an increase in the early 1990s and then decrease in BAAFFM, but a decrease in early 1990s and then increase in the distance to default. From 2000 to 2008, the BAAFFM is decreasing, but the distance to default is increasing. After 2008, the BAAFFM sharply increases and the distance to default sharply drops. Then, as BAAFFM decreases until 2020, the distance to default is increasing. Since DD and PD are inversely related, we can additionally see the increase in probability of default in the early 1990s along with the increase in BAAFFM those years. This similarly happens in 2001, and then 2008-2009.

[38]:
```python
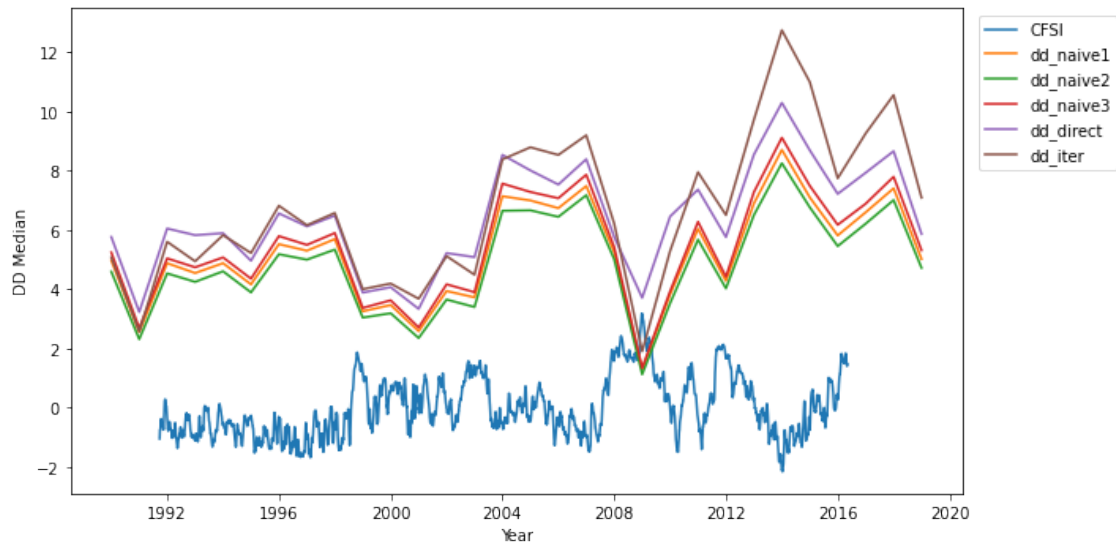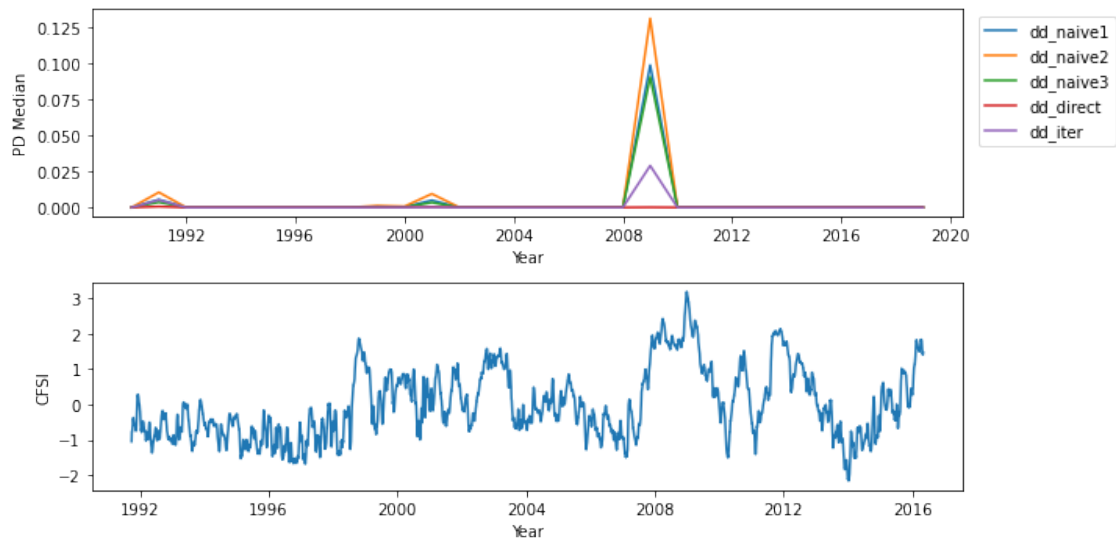# DD Median values with CFSI
plot_df(df_123_list[2], True, 5, 'DD Median', None, None, cfsi, fig_size=(10,
 ↪5),
        legend_loc='upper left', legend_bbox=(1.01, 1))
```

<Figure size 432x288 with 0 Axes>

```
[39]: # PD Median values with CFSI
      plot_df(df_123_median_pd, True, 5, 'PD Median', None, None, cfsi, sub_plot=True,
              fig_size=(10, 5), legend_loc='upper left', legend_bbox=(1.01, 1))
```

<Figure size 432x288 with 0 Axes>

The Cleveland Financial Stress Index and distance to default calculations appear to be inversely correlated. Periods of high financial stress, like 2009 show a high CFSI and low distance to default. Similarly, we see the high probability of default in 2009, mirrorring the increased CFSI.

14