hmm_sustainability2

June 13, 2021

[13]:
```
import numpy as np
import tensorflow as tf
import tensorflow_probability as tfp
import matplotlib.pyplot as plt
from google.colab import files
```

0.1 government approach (policy) to restrictions - mild, strong

0.2 restrictions - without, moderate, maximum (lockdown)

[2]:
```
tfd = tfp.distributions

initial_distribution = tfd.Categorical([0.0, 0.5, 0.5])

transition_distributions = [tfd.Categorical(probs=[[0.5, 0.5, 0.0],    # mild
                                                   [0.2, 0.6, 0.2],
                                                   [0.2, 0.7, 0.1]]),
                            tfd.Categorical(probs=[[0.0, 0.4, 0.6],    # strong
                                                   [0.1, 0.2, 0.7],
                                                   [0.0, 0.2, 0.8]])]

stable_trend_mean = 10076.9
fall_rate = np.array([0, 0.085, 0.17])
loc_value = (stable_trend_mean - stable_trend_mean * fall_rate).astype(np.
 ↪float32)
scale_value = (0.05 * loc_value).astype(np.float32)
observation_distribution = tfd.Normal(loc=loc_value, scale=scale_value)

num_periods = 7

values_mean = []
values_std = []

for transition_distribution in transition_distributions:
  model = tfd.HiddenMarkovModel(initial_distribution=initial_distribution,
                                transition_distribution=transition_distribution,
                                observation_distribution=observation_distribution,
                                num_steps=num_periods)
```

```
  values_mean.append(model.mean())
  values_std.append(model.stddev())
```

[15]:
```
np.array(values_mean)
#np.savetxt("values_mean.csv", np.array(values_mean), delimiter=",")
#files.download("values_mean.csv")
```

[15]:
```
array([[9091.064 , 9352.881 , 9345.772 , 9352.206 , 9353.279 , 9353.6875,
        9353.803 ],
       [9091.064 , 8640.719 , 8583.94  , 8574.933 , 8573.797 , 8573.616 ,
        8573.594 ]], dtype=float32)
```

[17]:
```
np.array(values_std)
#np.savetxt("values_std.csv", np.array(values_std), delimiter=",")
#files.download("values_std.csv")
```

[17]:
```
array([[801.9826 , 695.8299 , 712.85803, 711.47595, 711.69916, 711.70087,
        711.7079 ],
       [801.9826 , 635.6017 , 600.7586 , 592.4247 , 591.6242 , 591.45435,
        591.4383 ]], dtype=float32)
```

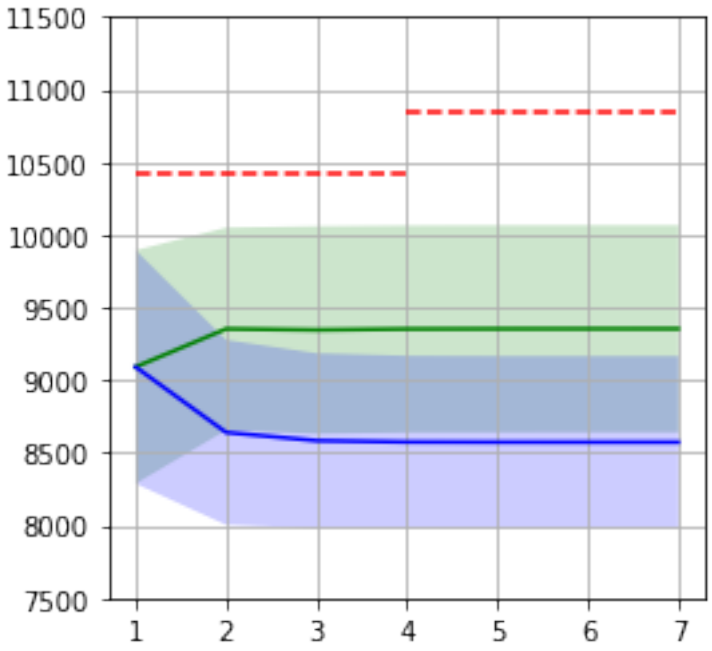
[9]:
```
print(np.array(values_mean[0][0:4]).mean())
print(np.array(values_mean[1][0:4]).mean())
```

```
9285.481
8722.664
```

[33]:
```
fig, ax = plt.subplots(figsize=(4,4))
plt.plot(range(1,8), values_mean[0], color='green')
plt.plot(range(1,8), values_mean[1], color='blue')

plt.fill_between(range(1,8), values_mean[0]+values_std[0],␣
 ↪values_mean[0]-values_std[0], facecolor='green', alpha=0.2)
plt.fill_between(range(1,8), values_mean[1]+values_std[1],␣
 ↪values_mean[1]-values_std[1], facecolor='blue', alpha=0.2)

plt.plot(range(1,5), [10425]*4, color='red', linestyle='--')
plt.plot(range(4,8), [10842]*4, color='red', linestyle='--')

ax.set_ylim([7500, 11500])
ax.grid()

#plt.savefig("abc.png")
#files.download("abc.png")
```

```
<IPython.core.display.Javascript object>
```

```
<IPython.core.display.Javascript object>
```


11500
11000
10500
10000
9500
9000
8500
8000
7500
1
2
3
4
5
6
7