Coherences Scores of LDA With Different Documents Scopes and NMF

In [1]:
```
from pickle import load#, dump
from my_lda_utils import my_lda_model
import pandas as pd
```

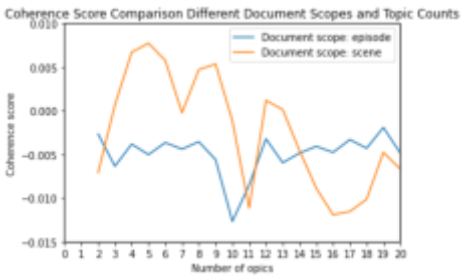
Number of Topics and Coherence LDA Document Scope Episode and Scene

In [65]:
```
with open("c_nmpi_coherence_approach3.pkl", "rb") as f1, open("c_nmpi_coherence_approach2.pkl", "rb") as f2:
    coherence_topic_counts_approach3 = load(f1)
    coherence_topic_counts_approach2 = load(f2)

compare_coherence = pd.merge(coherence_topic_counts_approach2['coherence'],coherence_topic_counts_approach3['coherence'],
                             suffixes = ('_2', '_3'),right_index=True, left_index=True)

compare_coherence.columns = ['Document scope: episode', 'Document scope: scene']
compare_coherence.plot(kind='line',
                  xticks=[x for x in range(21)],
                       yticks=[round(x*0.001, 5) for x in range(-15,15,5)],
                       xlim=[0,20],
                       xlabel="Number of opics",
                       ylabel="Coherence score",
                       title="Coherence Score Comparison Different Document Scopes and Topic Counts"
                      )
```

Out[65]: <AxesSubplot:title={'center':'Coherence Score Comparison Different Document Scopes and Topic Counts'}, xlabel='Number of opics', ylabel='Coherence score'>



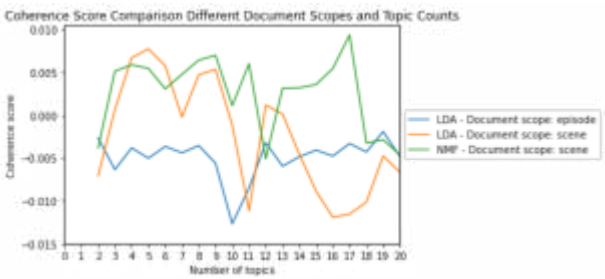
Number of Topics and General Coherence LDA Document Scope Episode and Scene and NMF Document Scope Scene

In [68]:
```
with open("c_nmpi_coherence_approach3.pkl", "rb") as f1, \
     open("c_nmpi_coherence_approach2.pkl", "rb") as f2, \
     open("c_nmpi_coherence_approach3_nmf.pkl", "rb") as f3:
    coherence_topic_counts_approach3 = load(f1)
    coherence_topic_counts_approach2 = load(f2)
    coherence_topic_counts_approach3_nmf = load(f3)


compare_coherence1 = pd.merge(coherence_topic_counts_approach2['coherence'],coherence_topic_counts_approach3['coherence'],
                              suffixes = ('_2', '_3'),right_index=True, left_index=True)
compare_coherence = pd.merge(compare_coherence1,coherence_topic_counts_approach3_nmf['coherence'], suffixes = ('', ''),
                             right_index=True, left_index=True)
compare_coherence.columns = ['LDA - Document scope: episode', 'LDA - Document scope: scene', 'NMF - Document scope: scene']

compare_coherence.plot(kind='line',
                  xticks=[x for x in range(21)],
                       yticks=[round(x*0.001, 5) for x in range(-15,15,5)],
                       xlim=[0,20],
                       xlabel="Number of topics",
                       ylabel="Coherence score",
                       title="Coherence Score Comparison Different Document Scopes and Topic Counts",
                      ).legend(loc='center left',bbox_to_anchor=(1.0, 0.5))
```

Out[68]: <matplotlib.legend.Legend at 0x20983c137f0>



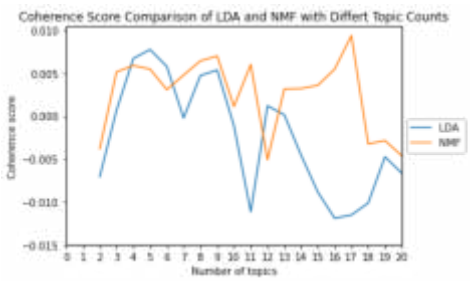
In [71]:

```
with open("c_nmpi_coherence_approach3.pkl", "rb") as f1, \
        open("c_nmpi_coherence_approach3_nmf.pkl", "rb") as f3:
    coherence_topic_counts_approach3 = load(f1)
    coherence_topic_counts_approach3_nmf = load(f3)


compare_coherence = pd.merge(coherence_topic_counts_approach3['coherence'],
                             coherence_topic_counts_approach3_nmf['coherence'],
                             suffixes = ('_LDA', '_NMF'),right_index=True, left_index=True)
compare_coherence.columns = ['LDA', 'NMF']

compare_coherence.plot(kind='line',
                  xticks=[x for x in range(21)],
                       yticks=[round(x*0.001, 5) for x in range(-15,15,5)],
                       xlim=[0,20],
                       xlabel="Number of topics",
                       ylabel="Coherence score",
                       title="Coherence Score Comparison of LDA and NMF with Differt Topic Counts",
                      ).legend(loc='center left',bbox_to_anchor=(1.0, 0.5))
```

Out[71]: <matplotlib.legend.Legend at 0x2098a09dbe0>



In [76]:

```
with open("c_nmpi_coherence_approach3.pkl", "rb") as f1, \
        open("c_nmpi_coherence_approach2.pkl", "rb") as f2, \
        open("c_nmpi_coherence_approach3_nmf.pkl", "rb") as f3:
    coherence_topic_counts_approach3 = load(f1)
    coherence_topic_counts_approach2 = load(f2)
    coherence_topic_counts_approach3_nmf = load(f3)


compare_coherence1 = pd.merge(coherence_topic_counts_approach2['coherence'],
                              coherence_topic_counts_approach3['coherence'],
                              suffixes = ('_2', '_3'),right_index=True, left_index=True)
compare_coherence = pd.merge(compare_coherence1,coherence_topic_counts_approach3_nmf['coherence'],
                             suffixes = ('', ''),right_index=True, left_index=True)
compare_coherence.columns = ['del', 'LDA', 'NMF']
compare_coherence['del'] = 10
compare_coherence.plot(kind='line',
                  xticks=[x for x in range(21)],
                       yticks=[round(x*0.001, 5) for x in range(-15,15,5)],
                       xlim=[0,20],
                       ylim=[-0.015,0.01],
                       xlabel="Number of topics",
                       ylabel="Coherence score",
                       title="Coherence Score Comparison Different Document Scopes and Topic Counts",
                      ).legend(loc='center left',bbox_to_anchor=(1.0, 0.5))
```

Out[76]: <matplotlib.legend.Legend at 0x2096fa7b040>