

syntheco

May 24, 2018

```
In [76]: %pylab inline
import seaborn as sns
sns.set(color_codes=True)
figsize(20,6)
import librosa
import librosa.display
```

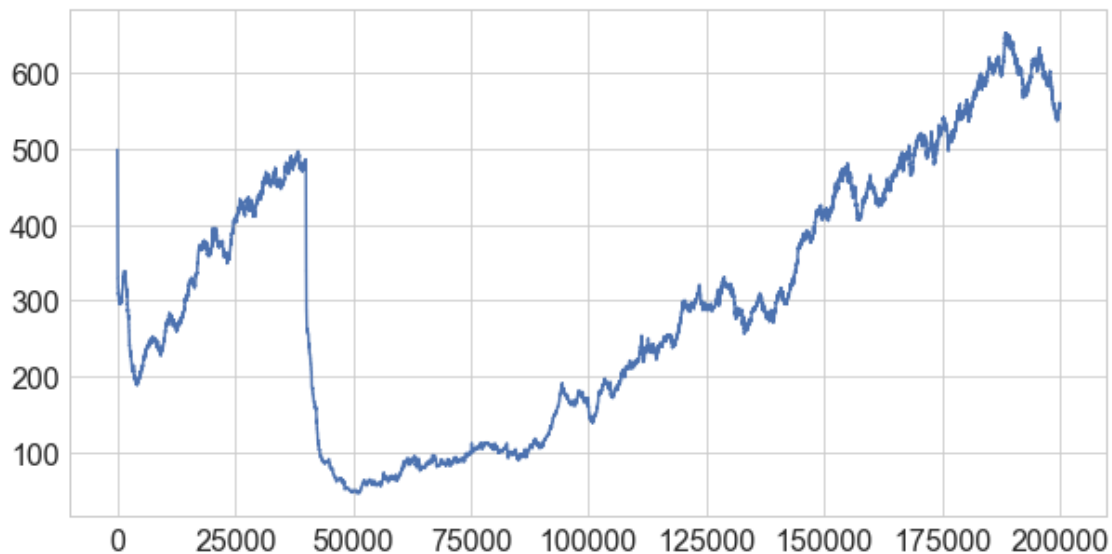
Populating the interactive namespace from numpy and matplotlib

```
/Volumes/LocalDataHD/virtualenvs/env3/lib/python3.6/site-packages/IPython/core/magics/pylab.py:1
`%matplotlib` prevents importing * from pylab and numpy
"\n`%matplotlib` prevents importing * from pylab and numpy"
```

```
In [90]: figsize(20,6)
sns.set_style("whitegrid")
sns.set_context("paper", font_scale=2)
```

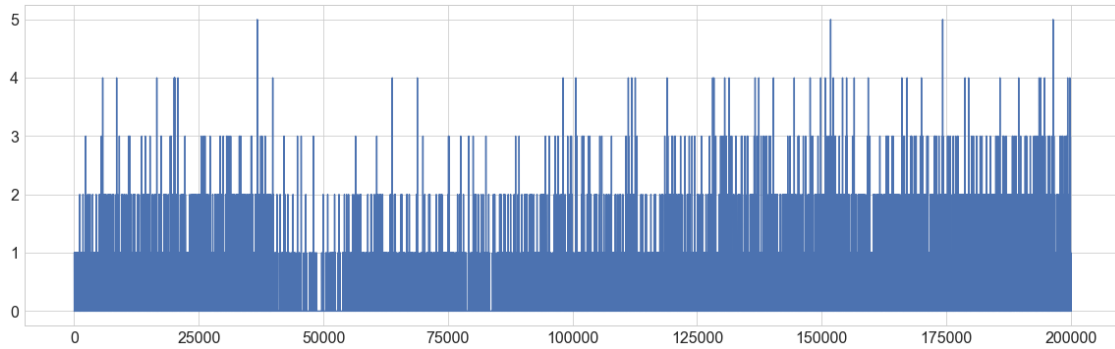
```
In [89]: pop = np.load("model1/popsizes.npy")
plot(pop)
```

```
Out[89]: [<matplotlib.lines.Line2D at 0x11846c0f0>]
```



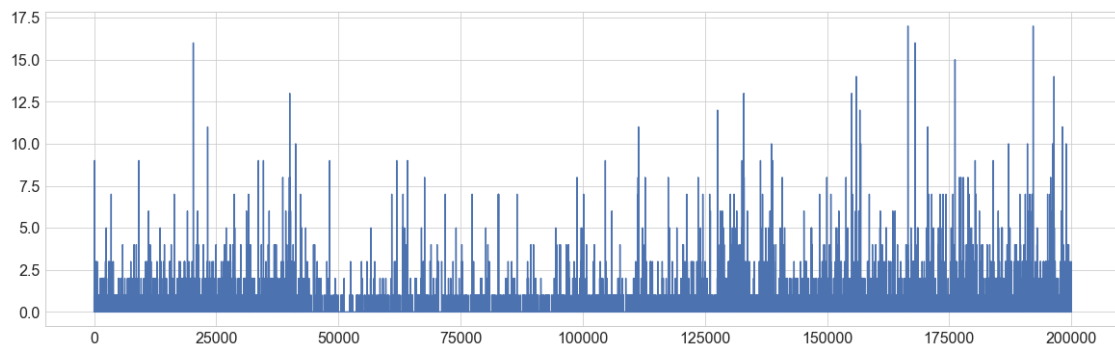
```
In [91]: births = np.load("model1/births.npy")
         plot(births)
```

```
Out [91]: [<matplotlib.lines.Line2D at 0x1189bfb70>]
```



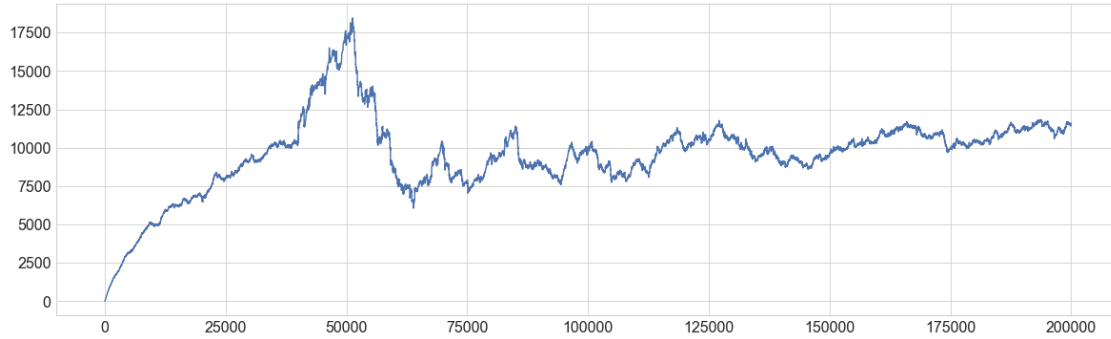
```
In [92]: deaths = np.load("model1/deaths.npy")
         plot(deaths)
```

```
Out [92]: [<matplotlib.lines.Line2D at 0x11735f438>]
```



```
In [93]: averageAge = np.load("model1/averageAge.npy")
         plot(averageAge)
```

```
Out [93]: [<matplotlib.lines.Line2D at 0x1233710f0>]
```



```
In [82]: import librosa
         w, sr = librosa.load("modell1/test.wav")
```

```
/Volumes/LocalDataHD/virtualenvs/env3/lib/python3.6/site-packages/resampy/core.py:90: FutureWarning
if not np.issubdtype(x.dtype, np.float):
```

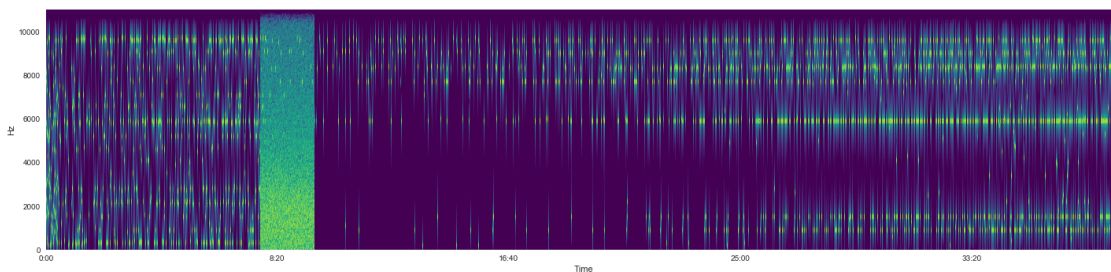
```
In [83]: w.shape
```

```
Out[83]: (51200000,)
```

```
In [84]: fft = librosa.stft(w, n_fft=512, hop_length = 512)
```

```
In [85]: import librosa.display
         figsize(20,5)
         librosa.display.specshow(librosa.amplitude_to_db(fft[:, :]),ref=np.max),
         y_axis='linear', x_axis='time', cmap='viridis')
         plt.tight_layout()
         plt.savefig("/tmp/modell1spectrogram.eps")
```

```
/Volumes/LocalDataHD/virtualenvs/env3/lib/python3.6/site-packages/librosa/display.py:656: FutureWarning
if np.issubdtype(data.dtype, np.complex):
```



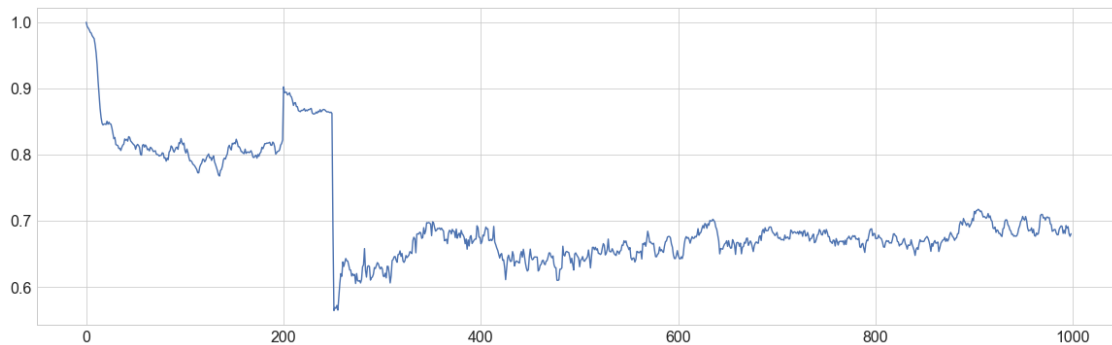
```

In [94]: #spectral entropy
fftdb = librosa.amplitude_to_db(fft, ref=np.max)
fftp = librosa.db_to_power(fftdb)
blocksize=100
blockcount = int(floor(fftp.shape[1] / blocksize))
ses = []
for i in range(blockcount-1):
    block = np.mean(fftp[:, i*blocksize:(i+1)*blocksize], axis=1)
    block = block / np.sum(block)
    se = np.sum(block * np.log(np.reciprocal(block)))
    ses.append(se)

plot(ses/np.max(ses))

```

Out [94]: [



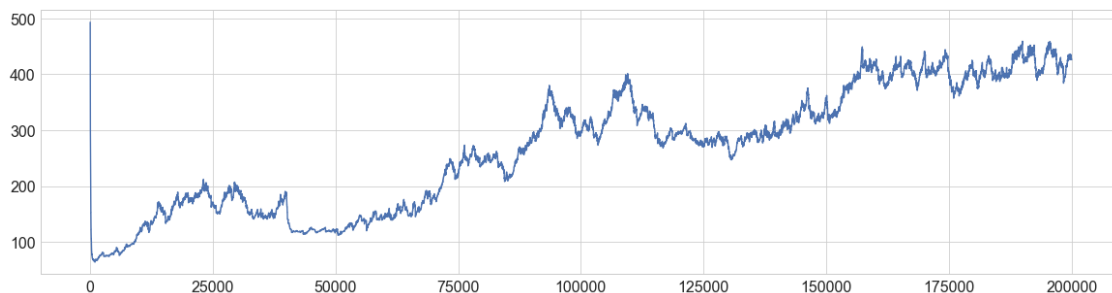
0.0.1 model 2

```

In [12]: pop2 = np.load("model2/popsize.npy")
plot(pop2)

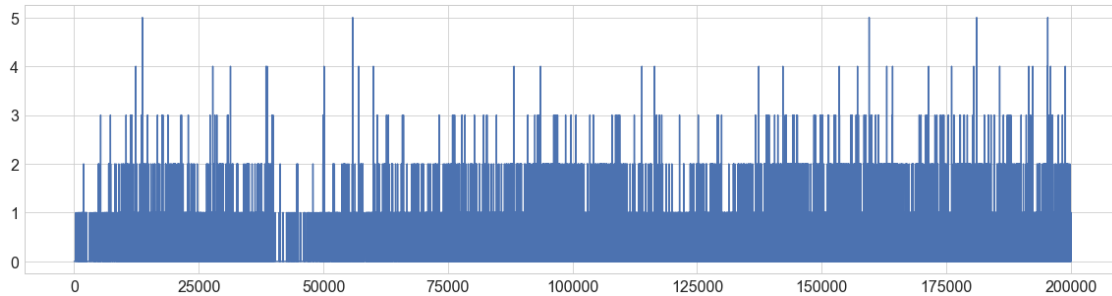
```

Out [12]: [



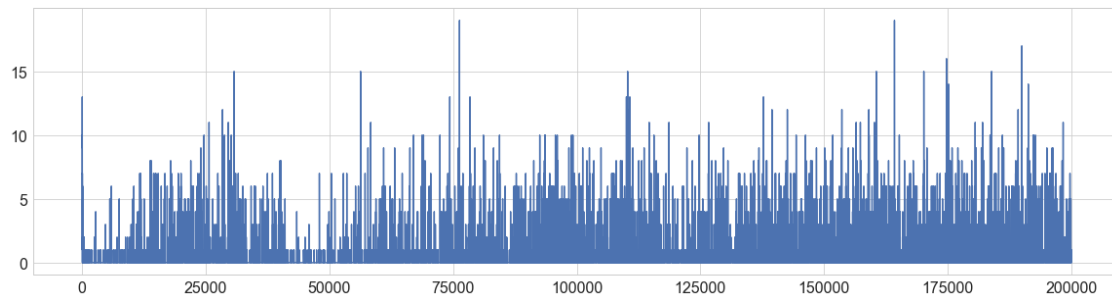
```
In [13]: births2 = np.load("model2/births.npy")
         plot(births2)
```

```
Out[13]: [<matplotlib.lines.Line2D at 0x111494518>]
```



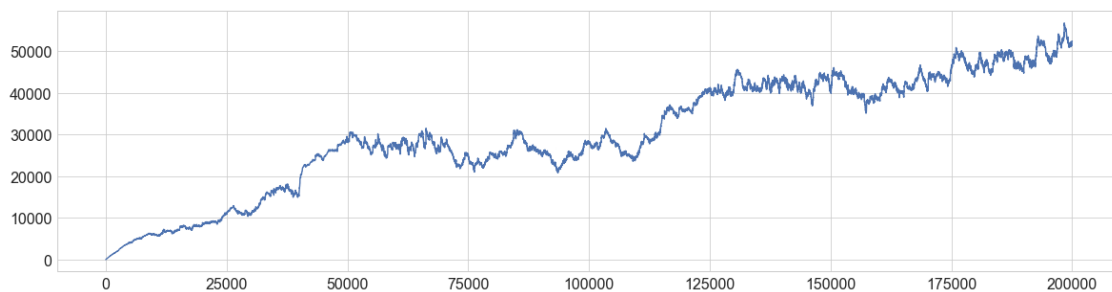
```
In [14]: deaths2 = np.load("model2/deaths.npy")
         plot(deaths2)
```

```
Out[14]: [<matplotlib.lines.Line2D at 0x111537630>]
```



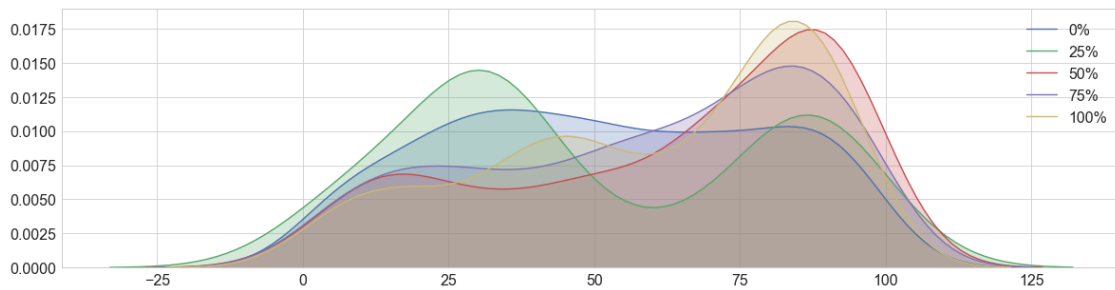
```
In [15]: averageAge2 = np.load("model2/averageAge.npy")
         # plot(averageAge)
         plot(averageAge2)
```

```
Out[15]: [<matplotlib.lines.Line2D at 0x11404aa20>]
```



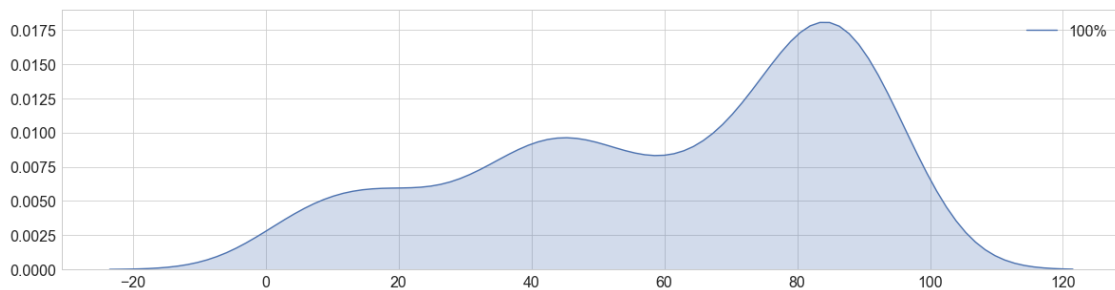
```
In [16]: periods = np.load("model2/periods.npy")
# imshow(periods.T * 10)
# plot(deaths)
# periods.shape
sns.kdeplot([x for x in periods[0] if x > 0], shade=True, label="0%")
sns.kdeplot([x for x in periods[int(periods.shape[0] * 0.25)] if x > 0], shade=True, label="25%")
sns.kdeplot([x for x in periods[int(periods.shape[0] * 0.5)] if x > 0], shade=True, label="50%")
sns.kdeplot([x for x in periods[int(periods.shape[0] * 0.75)] if x > 0], shade=True, label="75%")
sns.kdeplot([x for x in periods[int(periods.shape[0] - 1)] if x > 0], shade=True, label="100%")
plt.legend()
```

Out[16]: <matplotlib.legend.Legend at 0x114650b70>



```
In [17]: sns.kdeplot([x for x in periods[int(periods.shape[0] - 2)] if x > 0], shade=True, label="100%")
```

Out[17]: <matplotlib.axes._subplots.AxesSubplot at 0x11405d470>

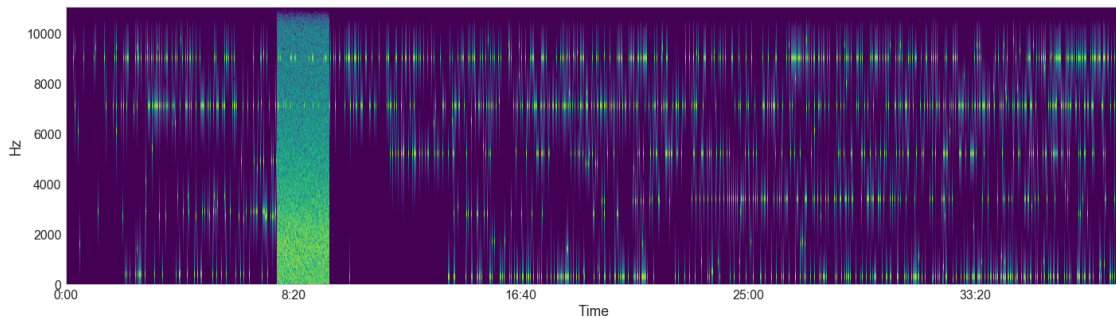


```
In [96]: w2, sr2 = librosa.load("model2/test.wav")

fft2 = librosa.stft(w2, n_fft=512, hop_length = 512)

librosa.display.specshow(librosa.amplitude_to_db(fft2[:, :, :], ref=np.max), y_axis='linear',
plt.tight_layout()
plt.savefig("/tmp/model2Spectrogram.eps")
```

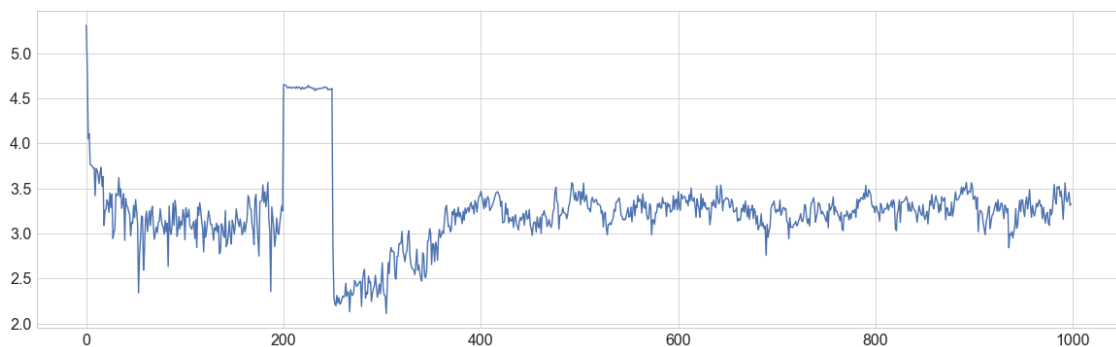
```
/Volumes/LocalDataHD/virtualenvs/env3/lib/python3.6/site-packages/resampy/core.py:90: FutureWarn
if not np.issubdtype(x.dtype, np.float):
/Volumes/LocalDataHD/virtualenvs/env3/lib/python3.6/site-packages/librosa/display.py:656: Future
if np.issubdtype(data.dtype, np.complex):
```



```
In [97]: fftdb2 = librosa.amplitude_to_db(fft2, ref=np.max)
fftp2 = librosa.db_to_power(fftdb2)
blockcount2 = int(floor(fftp2.shape[1] / blocksize))
ses2 = []
for i in range(blockcount2-1):
    block = np.mean(fftp2[:, i*blocksize:(i+1)*blocksize], axis=1)
    block = block / np.sum(block)
    se = np.sum(block * np.log(np.reciprocal(block)))
    ses2.append(se)
```

```
In [98]: plot(ses2)
```

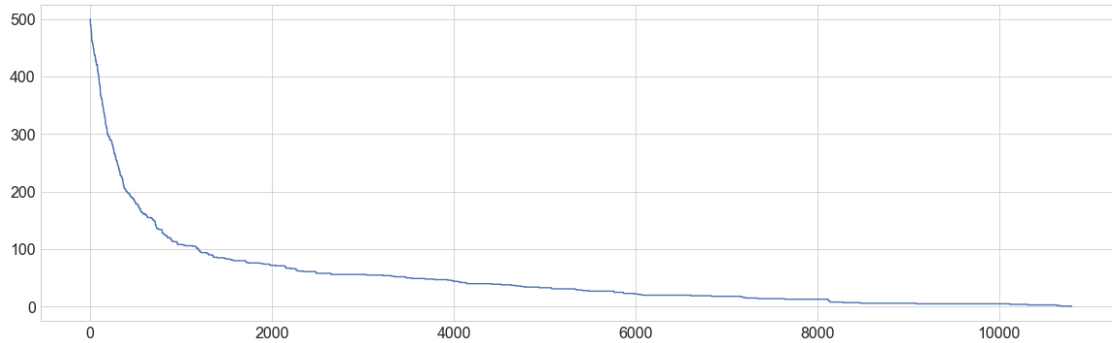
```
Out [98]: [<matplotlib.lines.Line2D at 0x1254ca9e8>]
```



0.0.2 Null Model - Minimal

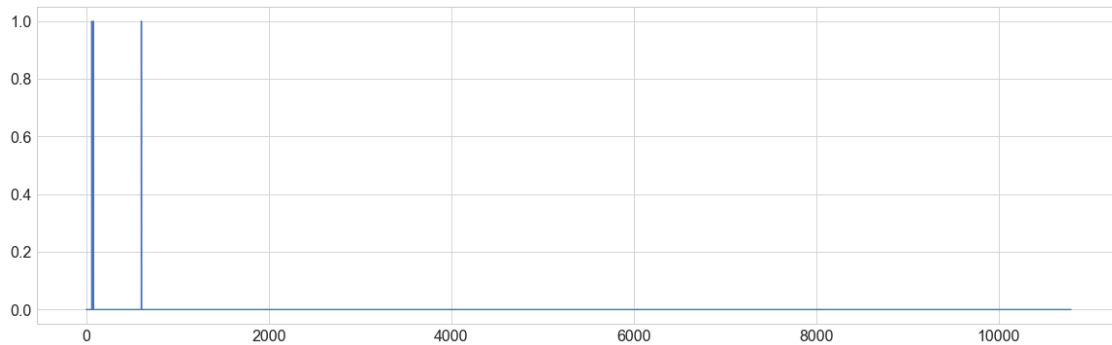
```
In [99]: pop3 = np.load("nullModel_minimal/popsize.npy")  
        plot(pop3)
```

```
Out[99]: [<matplotlib.lines.Line2D at 0x1174e0668>]
```



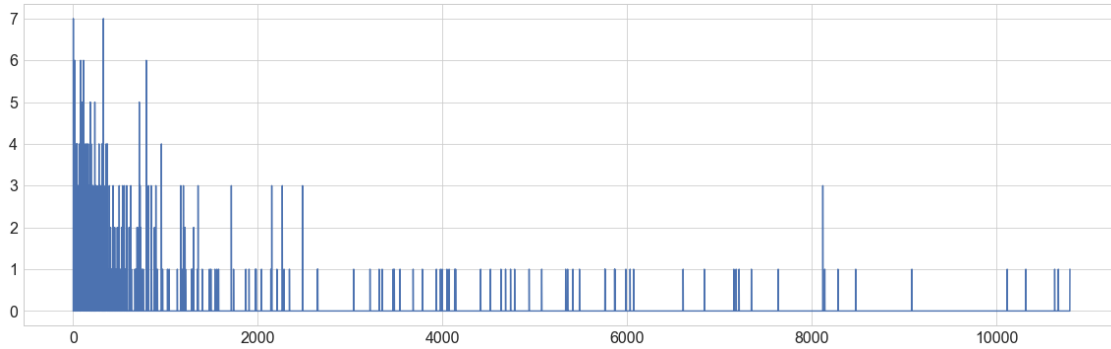
```
In [100]: births3 = np.load("nullModel_minimal/births.npy")  
         plot(births3)
```

```
Out[100]: [<matplotlib.lines.Line2D at 0x117b83518>]
```



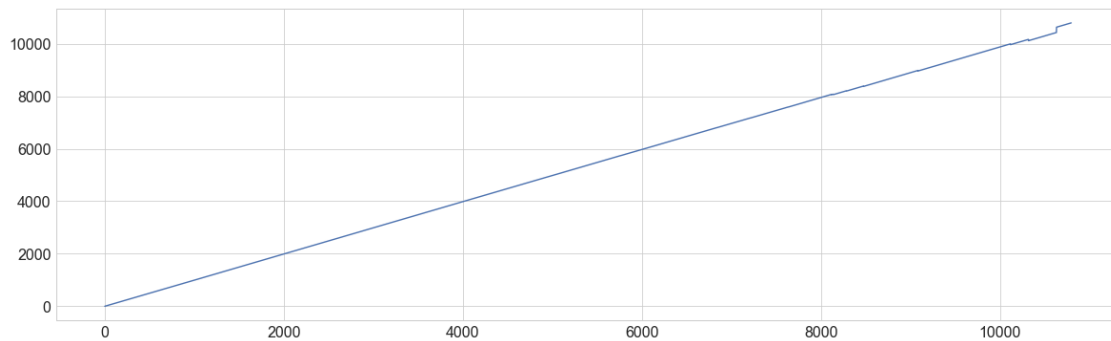
```
In [101]: deaths3 = np.load("nullModel_minimal/deaths.npy")  
         plot(deaths3)
```

```
Out[101]: [<matplotlib.lines.Line2D at 0x121e0b1d0>]
```

```
In [102]: averageAge3 = np.load("nullModel_minimal/averageAge.npy")
          # plot(averageAge)
          plot(averageAge3)
```

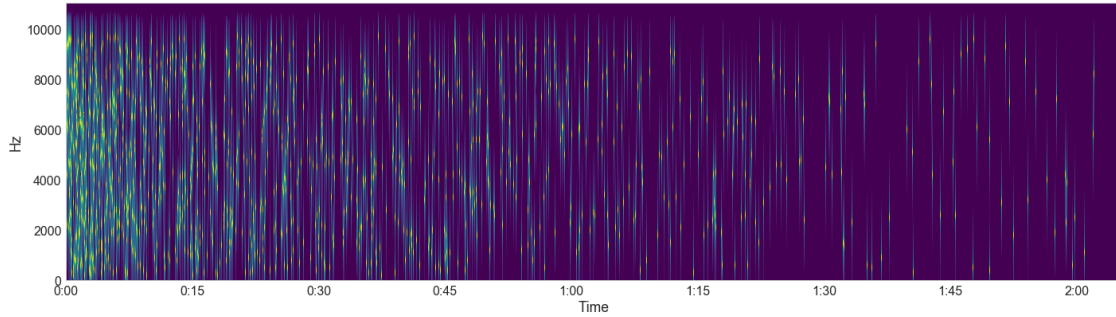
```
Out[102]: [<matplotlib.lines.Line2D at 0x121ed0b38>]
```



```
In [103]: w3, sr3 = librosa.load("nullModel_minimal/test.wav")

          fft3 = librosa.stft(w3, n_fft=512, hop_length = 512)
          librosa.display.specshow(librosa.amplitude_to_db(fft3[:,:],ref=np.max),y_axis='linear')
          plt.tight_layout()
          plt.savefig("/tmp/model3Spectrogram.eps")
```

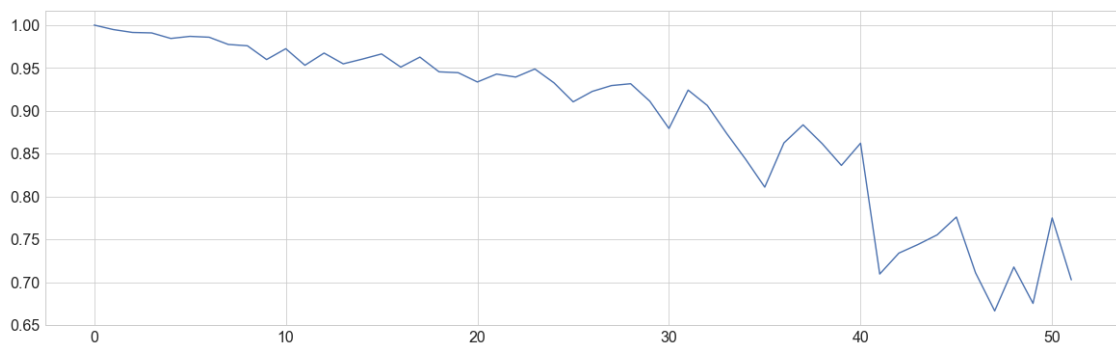
```
/Volumes/LocalDataHD/virtualenvs/env3/lib/python3.6/site-packages/resampy/core.py:90: FutureWarning
  if not np.issubdtype(x.dtype, np.float):
/Volumes/LocalDataHD/virtualenvs/env3/lib/python3.6/site-packages/librosa/display.py:656: FutureWarning
  if np.issubdtype(data.dtype, np.complex):
```



```
In [104]: #spectral entropy
fftdb3 = librosa.amplitude_to_db(fft3, ref=np.max)
fftp3 = librosa.db_to_power(fftdb3)
blocksize=100
blockcount3 = int(floor(fftp3.shape[1] / blocksize))
ses3 = []
for i in range(blockcount3-1):
    block = np.mean(fftp3[:, i*blocksize:(i+1)*blocksize], axis=1)
    block = block / np.sum(block)
    se = np.sum(block * np.log(np.reciprocal(block)))
    ses3.append(se)

plot(ses3/np.max(ses3))
```

Out[104]: [[matplotlib.lines.Line2D](#) at 0x12009db70>]



0.0.3 Null Model - Model 2 Parameters

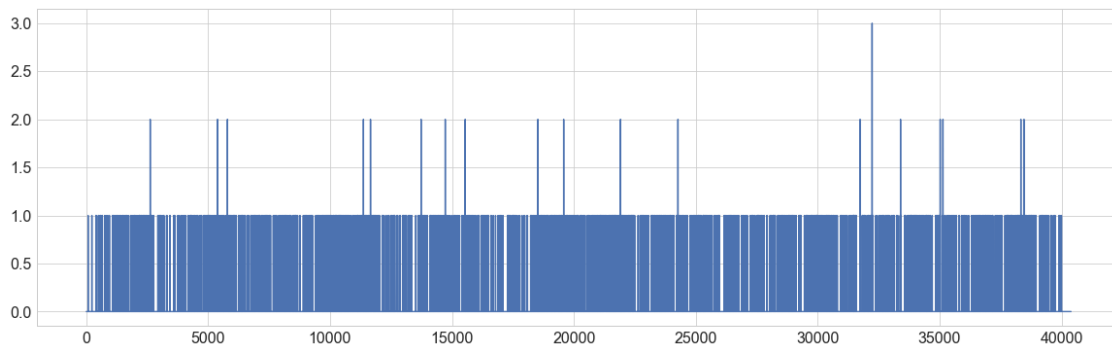
```
In [32]: figsize(20,6)
pop4 = np.load("nullModel/popsize.npy")
plot(pop4)
```

Out[32]: [[matplotlib.lines.Line2D](#) at 0x129fe7b70>]



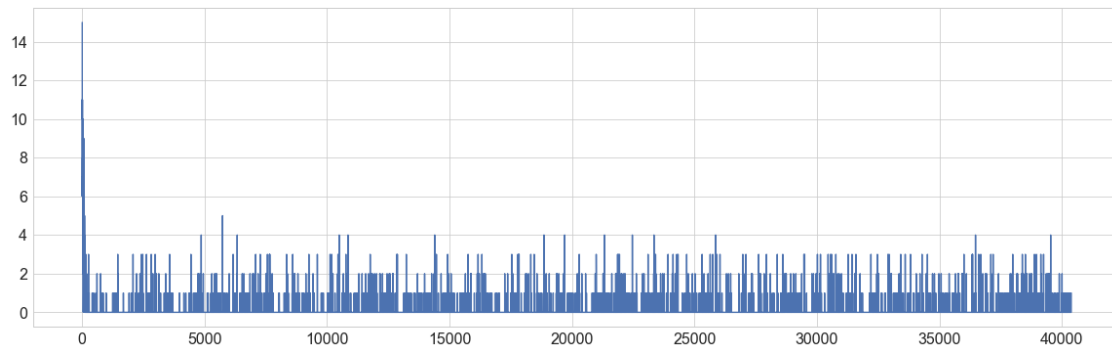
```
In [33]: births4 = np.load("nullModel/births.npy")  
         plot(births4)
```

```
Out[33]: [<matplotlib.lines.Line2D at 0x12a218f60>]
```



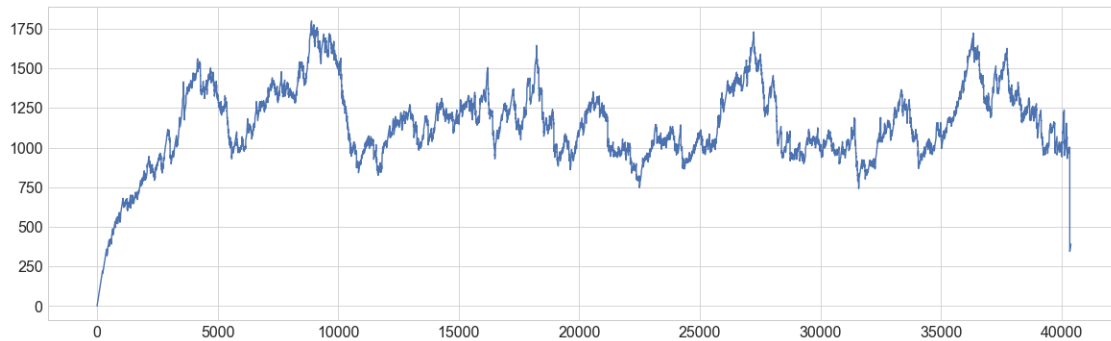
```
In [34]: deaths4 = np.load("nullModel/deaths.npy")  
         plot(deaths4)
```

```
Out[34]: [<matplotlib.lines.Line2D at 0x12a7dfac8>]
```



```
In [35]: averageAge4 = np.load("nullModel/averageAge.npy")
# plot(averageAge)
plot(averageAge4)
```

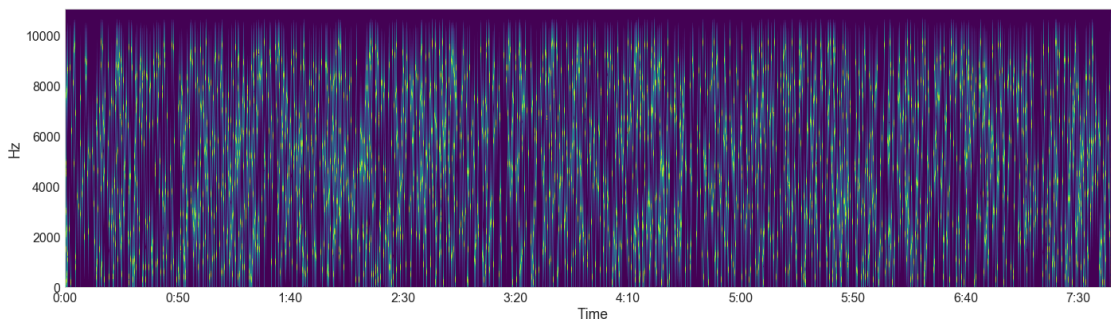
Out[35]: [



```
In [105]: w4, sr4 = librosa.load("nullModel/test.wav")
```

```
fft4 = librosa.stft(w4, n_fft=512, hop_length = 512)
librosa.display.specshow(librosa.amplitude_to_db(fft4[:,:],ref=np.max),y_axis='linear')
plt.tight_layout()
plt.savefig("/tmp/model4Spectrogram.eps")
```

```
/Volumes/LocalDataHD/virtualenvs/env3/lib/python3.6/site-packages/resampy/core.py:90: FutureWarning
if not np.issubdtype(x.dtype, np.float):
/Volumes/LocalDataHD/virtualenvs/env3/lib/python3.6/site-packages/librosa/display.py:656: FutureWarning
if np.issubdtype(data.dtype, np.complex):
```



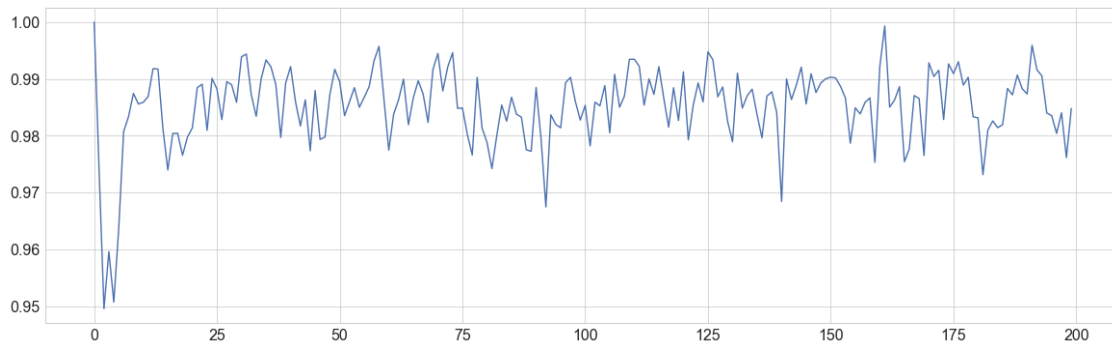
```

In [106]: #spectral entropy
fft4 = librosa.amplitude_to_db(fft4, ref=np.max)
fftp4 = librosa.db_to_power(fft4)
blocksize=100
blockcount4 = int(floor(fftp4.shape[1] / blocksize))
ses4 = []
for i in range(blockcount4-1):
    block = np.mean(fftp4[:, i*blocksize:(i+1)*blocksize], axis=1)
    block = block / np.sum(block)
    se = np.sum(block * np.log(np.reciprocal(block)))
    ses4.append(se)

plot(ses4/np.max(ses4))

```

Out[106]: [

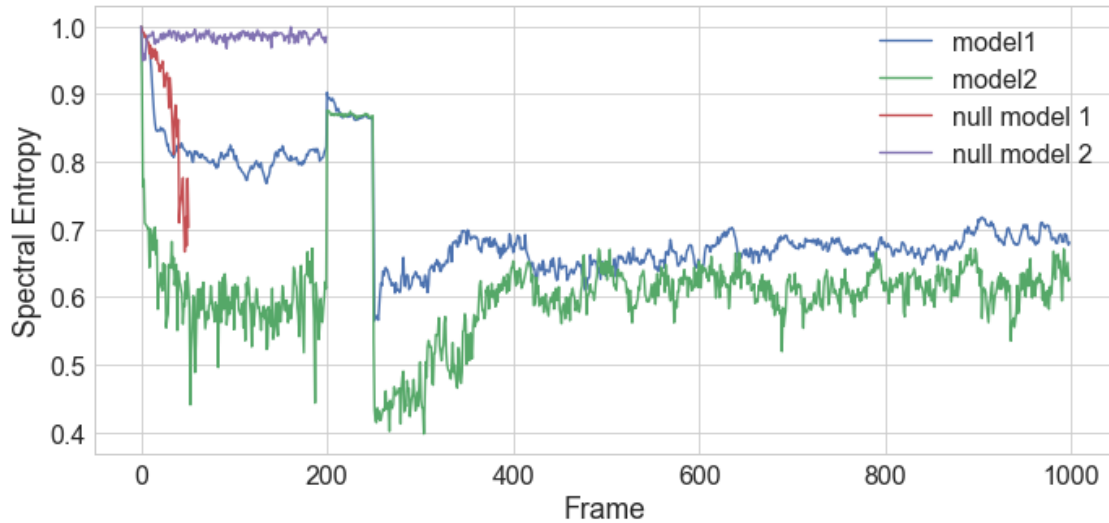


0.0.4 Spectral Entropy

```

In [107]: figsize(10,5)
fig = plt.figure()
ax = fig.add_subplot(1,1,1)
plot(ses / np.max(ses), label="model1")
plot(ses2 / np.max(ses2), label="model2")
plot(ses3 / np.max(ses3), label="null model 1")
plot(ses4 / np.max(ses4), label="null model 2")
plt.legend()
ax.set_xlabel("Frame")
ax.set_ylabel("Spectral Entropy")
plt.tight_layout()
plt.savefig('/tmp/spectralEntropy.eps')

```



In [41]: `### LZ Complexity`

```
from lempel_ziv_complexity import lempel_ziv_complexity
```

Info: numba.jit seems to be available.

In [68]: `lempel_ziv_complexity([0,0,0,0])`

Out[68]: 2

In [73]: `lzc = []`

```
for i in range(blockcount-1):
    block = np.mean(fftp[:, i*blocksize:(i+1)*blocksize], axis=1)
    blockmean = np.median(block)
    block = [1 if x else 0 for x in block > blockmean]
    lzc.append(lempel_ziv_complexity(block))
```

`lzc2 = []`

```
for i in range(blockcount2-1):
    block = np.mean(fftp2[:, i*blocksize:(i+1)*blocksize], axis=1)
    blockmean = np.median(block)
    block = [1 if x else 0 for x in block > blockmean]
    lzc2.append(lempel_ziv_complexity(block))
```

`lzc3 = []`

```
for i in range(blockcount3-1):
    block = np.mean(fftp3[:, i*blocksize:(i+1)*blocksize], axis=1)
    blockmean = np.median(block)
    block = [1 if x else 0 for x in block > blockmean]
```

```

lzc3.append(lempel_ziv_complexity(block))

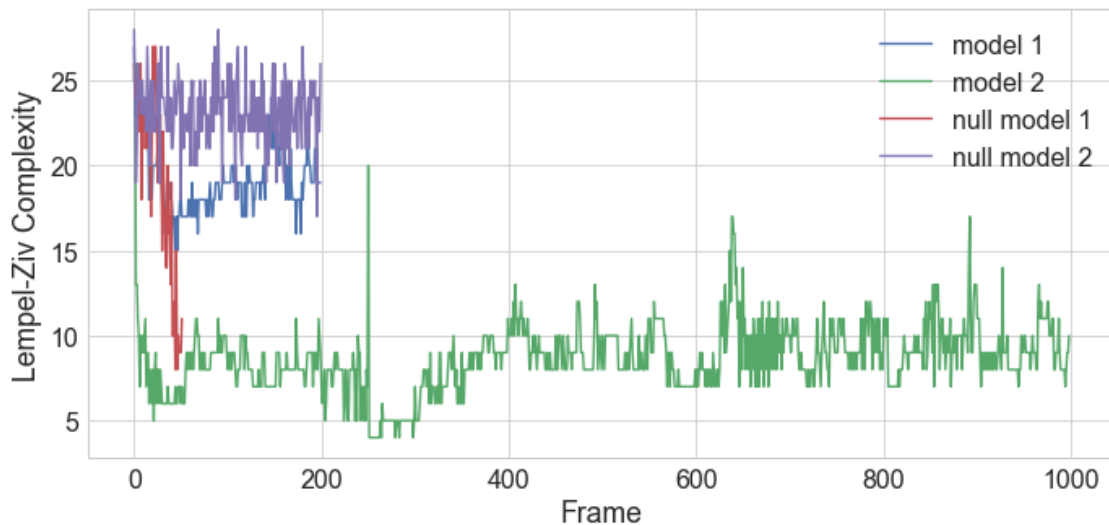
lzc4 = []
for i in range(blockcount4-1):
    block = np.mean(fftp4[:, i*blocksize:(i+1)*blocksize], axis=1)
    blockmean = np.median(block)
    block = [1 if x else 0 for x in block > blockmean]
    lzc4.append(lempel_ziv_complexity(block))

```

```

In [74]: figsize(10,5)
fig = plt.figure()
ax = fig.add_subplot(1,1,1)
plot(lzc, label="model 1")
plot(lzc2, label="model 2")
plot(lzc3, label="null model 1")
plot(lzc4, label="null model 2")
plt.legend()
ax.set_xlabel("Frame")
ax.set_ylabel("Lempel-Ziv Complexity")
plt.tight_layout()
plt.savefig("/tmp/lzcomplexity.eps")

```



```

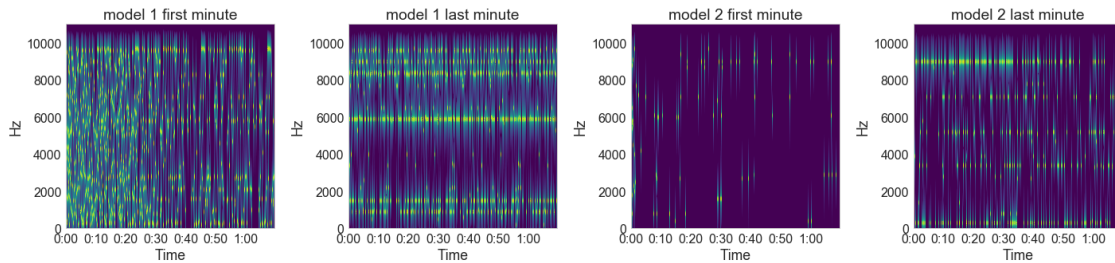
In [59]: figsize(20,5)
ax = subplot(1,4,1)
ax.set_title("model 1 first minute")
librosa.display.specshow(librosa.amplitude_to_db(fft[:, :3000], ref=np.max), y_axis='linea
ax = subplot(1,4,2)
ax.set_title("model 1 last minute")
librosa.display.specshow(librosa.amplitude_to_db(fft[:, -3000:], ref=np.max), y_axis='lin

```

```

ax = subplot(1,4,3)
ax.set_title("model 2 first minute")
librosa.display.specshow(librosa.amplitude_to_db(fft2[:, :3000],ref=np.max),y_axis='lin
ax = subplot(1,4,4)
ax.set_title("model 2 last minute")
librosa.display.specshow(librosa.amplitude_to_db(fft2[:, -3000:],ref=np.max),y_axis='li
plt.tight_layout()
plt.savefig("/tmp/specHighlights.eps")

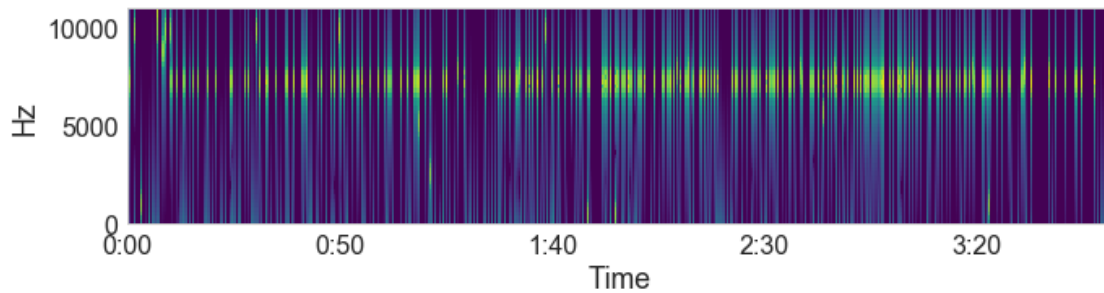
```



```

In [68]: figsize(10,3)
librosa.display.specshow(librosa.amplitude_to_db(fft2[-80:-30, -10000:],ref=np.max),
y_axis='linear', x_axis='time', cmap='viridis')
plt.tight_layout()
plt.savefig("/tmp/singleBandModel2.eps")

```



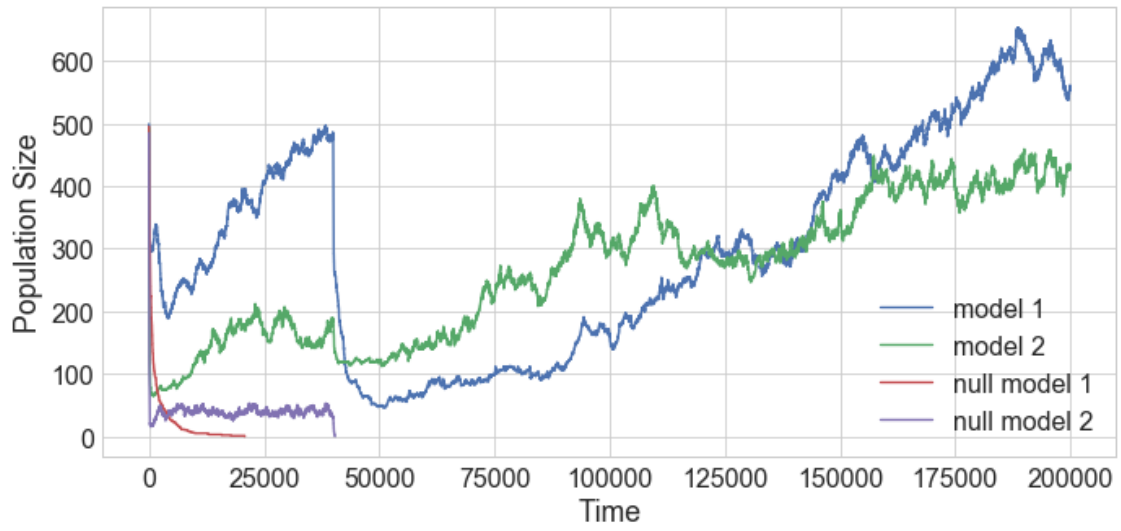
```

In [50]: figsize(10,5)
fig = plt.figure()
ax = fig.add_subplot(1,1,1)
plot(pop, label="model 1")
plot(pop2, label="model 2")
plot(pop3, label="null model 1")
plot(pop4, label="null model 2")
plt.legend()
ax.set_xlabel("Time")
ax.set_ylabel("Population Size")

```



```
plt.tight_layout()
plt.savefig("/tmp/popsize.eps")
```



In []: