Building a Guitar Model

Using the mean and the two most significant principal components from the Guitar Ensemble Analysis, let us try to resynthesize some notes:

In [1]:
```
%matplotlib inline
%load_ext autoreload
%autoreload 2
from pathlib import Path

# Enter the locations of the sample directories
CELLO_PATH  = Path("/home/lukas/BA/philharmonia-samples/cello")
GUITAR_PATH = Path("/home/lukas/BA/philharmonia-samples/guitar")

# Output directories for figures and wavfiles
GFX_PATH    = Path("/home/lukas/BA/report/gfx/")
WAVS_PATH   = Path("/home/lukas/BA/report/wavs/")
```

In [9]:
```
%%capture
# Initialization
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.decomposition import PCA
import librosa
import pya
import random

import principal_harmonics as ph

for path in [GFX_PATH, WAVS_PATH]:
    if path.exists() and not path.is_dir():
        raise NotADirectoryError(path)
    if not path.exists():
        path.mkdir()

pya.Aserver.startup_default_server()
```

In [3]:
```
guitar_mean = np.array([-36.43984264964854, -51.664528678171386, -55.677489302533225,
-71.65396272017958, -82.31237384797045, -88.0427329089259, -94.10093887744699,
-108.39540512034444, -107.53007814423783, -116.63714602859561,
-122.92400218020134, -130.0303821050933, -140.78434593992978,
-146.85556611583854, -169.32807134082955, -152.54990008177074,
-160.44719735111465, -158.17550468968184, -164.8131434551341,
-161.44591463015033, -171.94592995794108, -174.27594956518666,
-181.64454522416514, -174.16121457308006, -177.56937711757956,
-190.42113069159478, -177.5267830139337, -177.86936808340218,
-189.10382880072908, -192.01223955404384, -195.88168762869438,
-198.74440189104683, -194.31476819854518, -206.25384111878742,
-196.8940350256156, -206.76351034416754, -200.4398405068728,
-203.02302172680734, -197.5478880225631])
guitar_v1 = np.array([-0.012715285720582314, -0.03568272707395337, -0.05917673652472398,
-0.054597145039831846, -0.05565559330822718, -0.07064543180886329,
-0.0830978112115738, -0.11357706372802277, -0.10830557035811642,
-0.12228026141349384, -0.1379383005077171, -0.160821457996428,
-0.1739221045557159, -0.20689063203457234, -0.24357883958690646,
-0.2067270689632965, -0.21473820441275243, -0.21150574777624256,
-0.2065052127188612, -0.1966206219187784, -0.20462415961070007,
-0.21207193422301956, -0.2406621357486986, -0.2091606600830718,
-0.2422426527237797, -0.17138595965605907, -0.1655240179060535,
-0.1582895015938377, -0.14013936590706877, -0.1642693478119524,
-0.16905228226243807, -0.15679039795986374, -0.17317717519520925,
-0.13483156740442775, -0.12088329663239958, -0.10695274581228711,
-0.1575815287101677, -0.08854726603819016, -0.09821121417376827])
guitar_v2 = np.array([0.023152901944585347, 0.04484975732645691, 0.0439636589187137,
0.14560325826001086, 0.16683810399362042, 0.17307609234831717,
0.18144248618291325, 0.21913676647739933, 0.21496946956367619,
0.22179616189166146, 0.23821755735390127, 0.24734409922977646,
0.2185372741965388, 0.1433467934394349, 0.27703146764623243,
0.15150463641658773, 0.0556522317964786, 0.0654568323251394,
0.019980204585641488, 0.009763508163730929, 0.002906767219917991,
-0.028690639764002636, 0.042010057019852845, -0.15798744471372134,
-0.19443068944899675, -0.2095763940603465, -0.10385586531536298,
-0.06148940366018418, -0.09849973436072594, -0.12406337518577557,
-0.22567757517540898, -0.20044425209396582, -0.2416282575081171,
-0.15958029979579438, -0.12164087013475068, -0.13077925526939554,
-0.25881596954136343, -0.10426242904727236, -0.11275646423997568])
```

In [4]:
```
T     = 600
ts    = np.arange(T)
alpha = np.linspace(200, 1200, T)
beta  = 10 * np.sin(2*np.pi*0.05*ts)
timbres = guitar_mean.reshape(1, -1) + alpha.reshape(-1, 1) * guitar_v1.reshape(1, -1) + beta.reshape(-1, 1) * guit
guitar_asig = ph.pvoc.additive_resynth(freqs=880, ampls=pya.dbamp(timbres))
```

```
(1, 1) (1, 1) (600, 39)
```

```
ALSA lib pcm.c:8545:(snd_pcm_recover) underrun occurred
ALSA lib pcm.c:8545:(snd_pcm_recover) underrun occurred
```

In [5]:
```
guitar_asig = guitar_asig.remove_DC().fade_out(0.01).norm()
guitar_asig.save_wavfile(str(WAVS_PATH / 'ch5_guitar-model.wav'))
guitar_asig.play()
```

Out[5]:
```
Asig('_DCfree_fadeout_normalised'): 1 x 153344 @ 44100Hz = 3.477s cn=['0']
```

Building a Cello Model

In [6]:
```
cello_mean = np.array([-17.6037141607317, -21.36269526535325, -25.820114668799988,
-28.93254128427611, -32.97426434039337, -35.954244789252286, -40.17267387809677,
-41.86925104870337, -43.489325099872595, -46.50952754434864,
-48.601348775893875, -49.870935228712085, -51.908327455840684,
-53.31725330222326, -54.05127316199581, -55.46554103594453, -57.041396770155174,
-57.47426092391135, -240.0, -240.0, -240.0, -240.0, -240.0, -240.0, -240.0,
-240.0, -240.0, -240.0, -240.0, -240.0, -240.0, -240.0, -240.0, -240.0, -240.0,
-240.0, -240.0, -240.0, -240.0])
cello_v1 = np.array([0.04875310516819997, -0.13290332740610777, -0.15273466251689738,
-0.14305406619167502, -0.19717969964314222, -0.14240288887145802,
-0.20773484289747735, -0.24818954124525888, -0.2740526831265613,
-0.24888434770746673, -0.21640231640768295, -0.20366312578960416,
-0.27434483497259804, -0.2630267743729077, -0.31099283350265655,
-0.3174446881721234, -0.3057514320948768, -0.3338870482209155, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0])
cello_v2 = np.array([-0.31848928285225175, 0.2797634845301894, 0.15376344281669546,
0.1693501755026938, -0.15552265354168954, -0.1137052520730419,
0.1572783002694583, -0.1060964861732803, -0.25519632343977605,
-0.38521401068877287, -0.3487016108922941, -0.3590617682097907,
-0.10424317369572175, 0.2264118325773046, 0.3250072784001583,
0.22247162962270856, 0.06142258616914249, 0.09903014830176982, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0])
```

In [7]:
```
T     = 600
ts    = np.arange(T)
alpha = 20 * np.sin(2*np.pi*0.03*ts)
beta  = 15 * np.sin(2*np.pi*0.03*ts + np.pi / 2)
timbres = cello_mean.reshape(1, -1) + alpha.reshape(-1, 1) * cello_v1.reshape(1, -1) + beta.reshape(-1, 1) * cello_
cello_asig = ph.pvoc.additive_resynth(freqs=440, ampls=pya.dbamp(timbres))
```

```
(1, 1) (1, 1) (600, 39)
ALSA lib pcm.c:8545:(snd_pcm_recover) underrun occurred
```

In [8]:
```
cello_asig = cello_asig.remove_DC().fade_in(0.01).fade_out(0.01).norm() * 0.1
cello_asig.save_wavfile(str(WAVS_PATH / 'ch5_cello-model.wav'))
cello_asig.play()
```

Out[8]:
```
Asig('_DCfree_fadein_fadeout_normalised_multiplied'): 1 x 153344 @ 44100Hz = 3.477s cn=['0']
```