lbc exploratory

December 3, 2020

1 Experimental analysis using PLS (projection onto latent structures) regression

[74]:
```
import pandas as pd
import numpy as np
from sklearn.manifold import TSNE
import matplotlib.pyplot as plt
from matplotlib import figure
from sklearn.cross_decomposition import PLSCanonical, PLSRegression, CCA
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import StandardScaler
```

[ ]:
```
```

[2]:
```
# Import LBC data
lbc_csv = pd.read_csv("./data/LBC_Olink.csv")
```

[3]:
```
lbc_csv.head()
```

[3]:
```
   Unnamed: 0         ID       IL8     VEGFA     MCP.3     CDCP1     CD244  \
0           1  LBC360002  0.375674 -2.993634  0.438218 -2.993635 -2.299404
1           2  LBC360003  0.350047 -1.578708  0.731644  1.514374  0.013987
2           3  LBC360004  0.050058  0.365395 -0.634440 -0.075025 -0.474514
3           4  LBC360006 -0.891912 -0.341155 -0.884787  0.141817  0.373100
4           5  LBC360007 -0.819190 -1.045383 -0.628572 -1.379373 -1.523092

        IL7       OPG  LAP.TGF.beta.1  …  ICVc_nawm_mm3_w2  \
0  1.055876 -2.993634       -1.457822  …          0.342898
1 -0.679173 -0.111247       -0.682203  …          0.337526
2  0.574528 -1.254145       -0.303159  …          0.324664
3  0.760070 -0.361619        0.033220  …               NaN
4 -1.411016  0.257021       -1.464773  …          0.359640

   ICVc_brain_mm3_w2       wmh  ICVc_wmh_mm3_w2       gFA       gMD         g  \
0           0.680727  8.102586         0.000005  2.040635 -0.885258  2.236336
1           0.682304  9.292565         0.000007 -0.298203  0.965785  0.769793
```

```
2           0.688646  7.464510         0.000006  2.108851 -0.904939  0.163274
3                NaN  7.978311         0.000006  2.265567 -0.854490  2.006747
4           0.696776  8.936824         0.000006  0.404623  0.437485  0.197240

   visuospatial_ability  processing_speed  verbal_memory
0              1.808163          0.599615       1.390191
1              1.062659         -0.115145       0.887351
2              0.074846          0.141208      -0.352619
3              1.443611          1.411047       1.200256
4              0.275451          0.472795      -1.186996

[5 rows x 187 columns]
```

[84]:
```
# Variable label groups
proteins = ["IL8",
"VEGFA",
"MCP.3",
"CDCP1",
"CD244",
"IL7",
"OPG",
"LAP.TGF.beta.1",
"uPA",
"IL6",
"MCP.1",
"CXCL11",
"AXIN1",
"TRAIL",
"CXCL9",
"CST5",
"OSM",
"CXCL1",
"CCL4",
"CD6",
"SCF",
"IL18",
"SLAMF1",
"TGF.alpha",
"MCP.4",
"CCL11",
"TNFSF14",
"FGF.23",
"FGF.5",
"MMP.1",
"LIF.R",
"FGF.21",
"CCL19",
```

```
"IL.15RA",
"IL.10RB",
"IL.18R1",
"PD.L1",
"Beta.NGF",
"CXCL5",
"TRANCE",
"HGF",
"IL.12B",
"MMP.10",
"IL10",
"CCL23",
"CD5",
"CCL3",
"Flt3L",
"CXCL6",
"CXCL10",
"X4E.BP1",
"SIRT2",
"CCL28",
"DNER",
"EN.RAGE",
"CD40",
"FGF.19",
"MCP.2",
"CASP.8",
"CCL25",
"CX3CL1",
"TNFRSF9",
"NT.3",
"TWEAK",
"CCL20",
"ST1A1",
"STAMBP",
"ADA",
"TNFB",
"CSF.1"]
age = ["ageyears_w2"]
sex = ["sex"]
factors = [
    "ICVc_gm_mm3_w2"
]
```

[92]:
```
# Filling in missing values with column mean
# Use
imp = SimpleImputer(missing_values=np.nan, strategy='mean')
imp.fit(lbc_csv[proteins + age + sex])
```

```
lbc_proteins = imp.transform(lbc_csv[proteins + age + sex])
imp.fit(lbc_csv[factors])
lbc_factors = imp.transform(lbc_csv[factors])
```

[93]:
```
# Scale using statistical scoring
scaler = StandardScaler()
X = np.array(lbc_proteins)
scaler.fit(X)
X = scaler.transform(X)
Y = np.array(lbc_factors)
scaler.fit(Y)
Y = scaler.transform(Y)
print(X.shape)
print(Y.shape)
```

```
(758, 72)
(758, 1)
```

[97]:
```
# Fit PLS model
pls2 = PLSRegression(n_components=3)
pls2.fit(X, Y)
```

[97]:
```
PLSRegression(n_components=3)
```

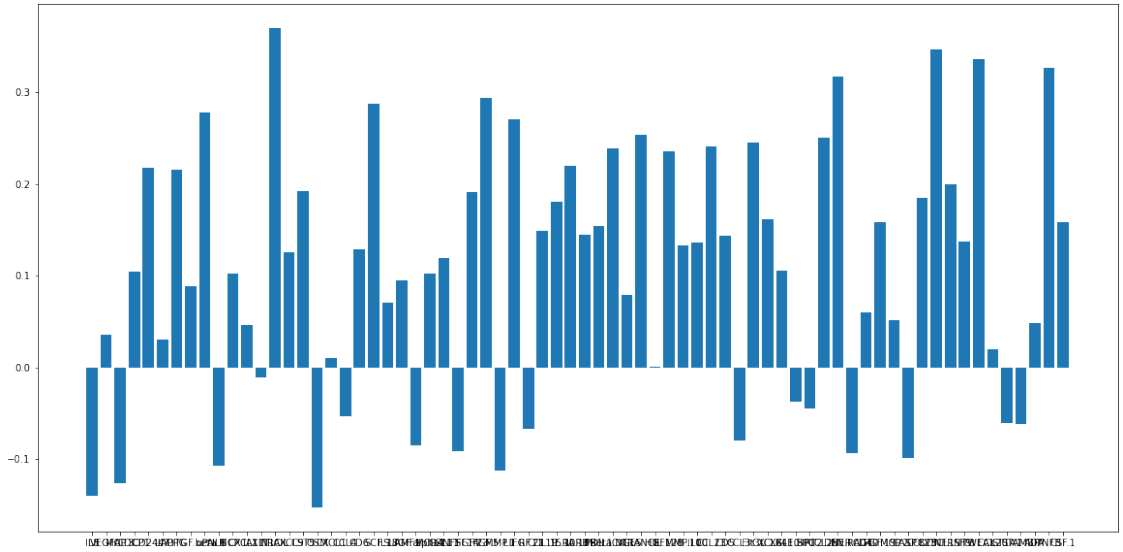
[101]:
```
# Plot PLS loadings (importance of variables on model)
# Y-axis shows relative contribution of protein to model

loadings = pls2.x_loadings_[:-2]
ind = np.arange(len(loadings))
plt.figure(figsize=(20,10))
plt.bar(ind, loadings[:,0])
plt.xticks(ind, proteins)
```

[101]:
```
([<matplotlib.axis.XTick at 0x7f80482a56a0>,
  <matplotlib.axis.XTick at 0x7f80482a5670>,
  <matplotlib.axis.XTick at 0x7f80482a2ca0>,
  <matplotlib.axis.XTick at 0x7f80486285e0>,
  <matplotlib.axis.XTick at 0x7f8048628af0>,
  <matplotlib.axis.XTick at 0x7f804862f040>,
  <matplotlib.axis.XTick at 0x7f804862f550>,
  <matplotlib.axis.XTick at 0x7f8048562910>,
  <matplotlib.axis.XTick at 0x7f804862f8b0>,
  <matplotlib.axis.XTick at 0x7f804862fdc0>,
  <matplotlib.axis.XTick at 0x7f8048637310>,
  <matplotlib.axis.XTick at 0x7f8048637820>,
  <matplotlib.axis.XTick at 0x7f8048637d30>,
  <matplotlib.axis.XTick at 0x7f804863d280>,
```

```
<matplotlib.axis.XTick at 0x7f804863d790>,
<matplotlib.axis.XTick at 0x7f804863dca0>,
<matplotlib.axis.XTick at 0x7f80486373a0>,
<matplotlib.axis.XTick at 0x7f80485628e0>,
<matplotlib.axis.XTick at 0x7f804863ddc0>,
<matplotlib.axis.XTick at 0x7f8048645310>,
<matplotlib.axis.XTick at 0x7f8048645820>,
<matplotlib.axis.XTick at 0x7f8048645d30>,
<matplotlib.axis.XTick at 0x7f804864b280>,
<matplotlib.axis.XTick at 0x7f804864b790>,
<matplotlib.axis.XTick at 0x7f804864bca0>,
<matplotlib.axis.XTick at 0x7f8048645550>,
<matplotlib.axis.XTick at 0x7f804862fa90>,
<matplotlib.axis.XTick at 0x7f804864b460>,
<matplotlib.axis.XTick at 0x7f8048652280>,
<matplotlib.axis.XTick at 0x7f8048652790>,
<matplotlib.axis.XTick at 0x7f8048652ca0>,
<matplotlib.axis.XTick at 0x7f80486561f0>,
<matplotlib.axis.XTick at 0x7f8048656700>,
<matplotlib.axis.XTick at 0x7f8048656c10>,
<matplotlib.axis.XTick at 0x7f8048656790>,
<matplotlib.axis.XTick at 0x7f8048652820>,
<matplotlib.axis.XTick at 0x7f804862ff10>,
<matplotlib.axis.XTick at 0x7f8048660340>,
<matplotlib.axis.XTick at 0x7f8048660850>,
<matplotlib.axis.XTick at 0x7f8048660d60>,
<matplotlib.axis.XTick at 0x7f80486652b0>,
<matplotlib.axis.XTick at 0x7f80486657c0>,
<matplotlib.axis.XTick at 0x7f8048665cd0>,
<matplotlib.axis.XTick at 0x7f80486654c0>,
<matplotlib.axis.XTick at 0x7f8048660550>,
<matplotlib.axis.XTick at 0x7f8048656250>,
<matplotlib.axis.XTick at 0x7f804866c3d0>,
<matplotlib.axis.XTick at 0x7f804866c8e0>,
<matplotlib.axis.XTick at 0x7f804866cdf0>,
<matplotlib.axis.XTick at 0x7f8048672340>,
<matplotlib.axis.XTick at 0x7f8048672850>,
<matplotlib.axis.XTick at 0x7f8048672d60>,
<matplotlib.axis.XTick at 0x7f8048672580>,
<matplotlib.axis.XTick at 0x7f804866c610>,
<matplotlib.axis.XTick at 0x7f8048660eb0>,
<matplotlib.axis.XTick at 0x7f8048679460>,
<matplotlib.axis.XTick at 0x7f8048679970>,
<matplotlib.axis.XTick at 0x7f8048679e80>,
<matplotlib.axis.XTick at 0x7f804867e3d0>,
<matplotlib.axis.XTick at 0x7f804867e8e0>,
<matplotlib.axis.XTick at 0x7f804867edf0>,
```

```
 <matplotlib.axis.XTick at 0x7f804867e610>,
 <matplotlib.axis.XTick at 0x7f80486796a0>,
 <matplotlib.axis.XTick at 0x7f804866cf70>,
 <matplotlib.axis.XTick at 0x7f80486854f0>,
 <matplotlib.axis.XTick at 0x7f8048685a00>,
 <matplotlib.axis.XTick at 0x7f8048685f10>,
 <matplotlib.axis.XTick at 0x7f804868c460>,
 <matplotlib.axis.XTick at 0x7f804868c970>,
 <matplotlib.axis.XTick at 0x7f804868ce80>],
[Text(0, 0, 'IL8'),
 Text(1, 0, 'VEGFA'),
 Text(2, 0, 'MCP.3'),
 Text(3, 0, 'CDCP1'),
 Text(4, 0, 'CD244'),
 Text(5, 0, 'IL7'),
 Text(6, 0, 'OPG'),
 Text(7, 0, 'LAP.TGF.beta.1'),
 Text(8, 0, 'uPA'),
 Text(9, 0, 'IL6'),
 Text(10, 0, 'MCP.1'),
 Text(11, 0, 'CXCL11'),
 Text(12, 0, 'AXIN1'),
 Text(13, 0, 'TRAIL'),
 Text(14, 0, 'CXCL9'),
 Text(15, 0, 'CST5'),
 Text(16, 0, 'OSM'),
 Text(17, 0, 'CXCL1'),
 Text(18, 0, 'CCL4'),
 Text(19, 0, 'CD6'),
 Text(20, 0, 'SCF'),
 Text(21, 0, 'IL18'),
 Text(22, 0, 'SLAMF1'),
 Text(23, 0, 'TGF.alpha'),
 Text(24, 0, 'MCP.4'),
 Text(25, 0, 'CCL11'),
 Text(26, 0, 'TNFSF14'),
 Text(27, 0, 'FGF.23'),
 Text(28, 0, 'FGF.5'),
 Text(29, 0, 'MMP.1'),
 Text(30, 0, 'LIF.R'),
 Text(31, 0, 'FGF.21'),
 Text(32, 0, 'CCL19'),
 Text(33, 0, 'IL.15RA'),
 Text(34, 0, 'IL.10RB'),
 Text(35, 0, 'IL.18R1'),
 Text(36, 0, 'PD.L1'),
 Text(37, 0, 'Beta.NGF'),
```

```
Text(38, 0, 'CXCL5'),
Text(39, 0, 'TRANCE'),
Text(40, 0, 'HGF'),
Text(41, 0, 'IL.12B'),
Text(42, 0, 'MMP.10'),
Text(43, 0, 'IL10'),
Text(44, 0, 'CCL23'),
Text(45, 0, 'CD5'),
Text(46, 0, 'CCL3'),
Text(47, 0, 'Flt3L'),
Text(48, 0, 'CXCL6'),
Text(49, 0, 'CXCL10'),
Text(50, 0, 'X4E.BP1'),
Text(51, 0, 'SIRT2'),
Text(52, 0, 'CCL28'),
Text(53, 0, 'DNER'),
Text(54, 0, 'EN.RAGE'),
Text(55, 0, 'CD40'),
Text(56, 0, 'FGF.19'),
Text(57, 0, 'MCP.2'),
Text(58, 0, 'CASP.8'),
Text(59, 0, 'CCL25'),
Text(60, 0, 'CX3CL1'),
Text(61, 0, 'TNFRSF9'),
Text(62, 0, 'NT.3'),
Text(63, 0, 'TWEAK'),
Text(64, 0, 'CCL20'),
Text(65, 0, 'ST1A1'),
Text(66, 0, 'STAMBP'),
Text(67, 0, 'ADA'),
Text(68, 0, 'TNFB'),
Text(69, 0, 'CSF.1')])
```



[102]:
```
subset = lbc_csv[proteins]
X = np.array(subset)
X.shape
```

[102]: (758, 70)

[129]:
```
# Fit TSNE model
X_emb = TSNE(n_components=2, perplexity=1, learning_rate=100, n_iter=1000,␣
 ↪init='pca').fit_transform(X)
X_emb.shape
```

[129]: (758, 2)

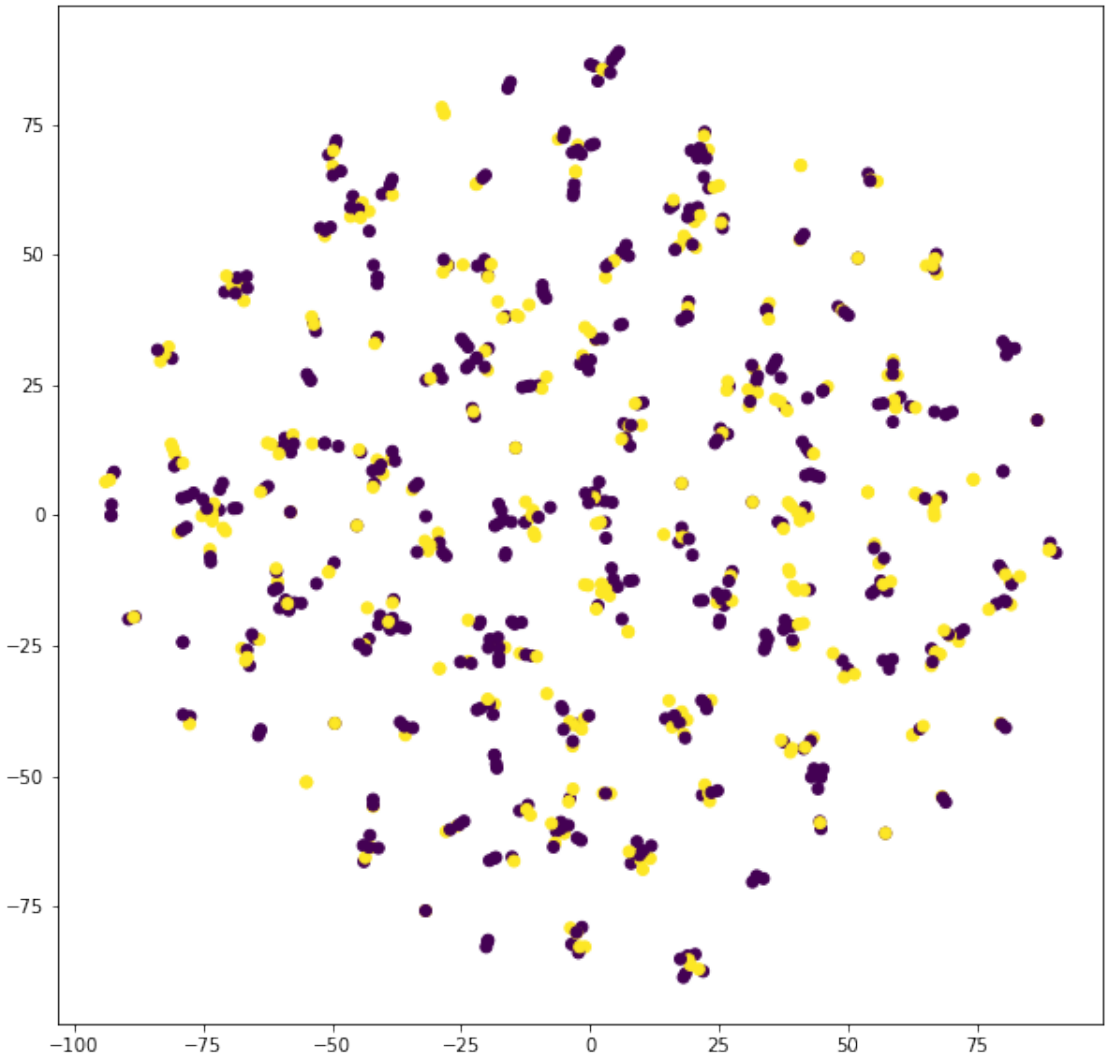
[125]:
```
# Colour labels based on risk factors
c_smoke = lbc_csv['smokprev_w2']
c_smokenow = lbc_csv['smokcurr_w2']
c_al = lbc_csv['alcfreq_w2']
c_sex = lbc_csv['sex']
c_gout = lbc_csv['gout_w1']
c_age = lbc_csv['agedays_w2']
c_bmi = lbc_csv['bmi_w2']
c_diab = lbc_csv['diab_w2']
c_chol = lbc_csv['hichol_w2']
c_six = lbc_csv['sixmwk_w2']
c_cog = lbc_csv['g']
c_all = (1+c_smoke * 2) + (2+ c_al * 3)
```

[112]:
```
diab = lbc_csv['diab']
cvdhist = lbc_csv['cvdhist'] * 10
hichol = lbc_csv['hichol'] * 20
stroke = lbc_csv['stroke'] * 30
parkin = lbc_csv['parkin'] * 40
hibp = lbc_csv['hibp'] * 50
demente = lbc_csv['demente'] * 60
code = diab + cvdhist + hichol + stroke + parkin + hibp + demente
risk = code > 0
```

[107]:
```
gmr = lbc_csv['gmIcv_ratio_w2']
ratio = lbc_csv['brainIcv_ratio_w2']
gmv = lbc_csv['ICVc_gm_mm3_w2']
```

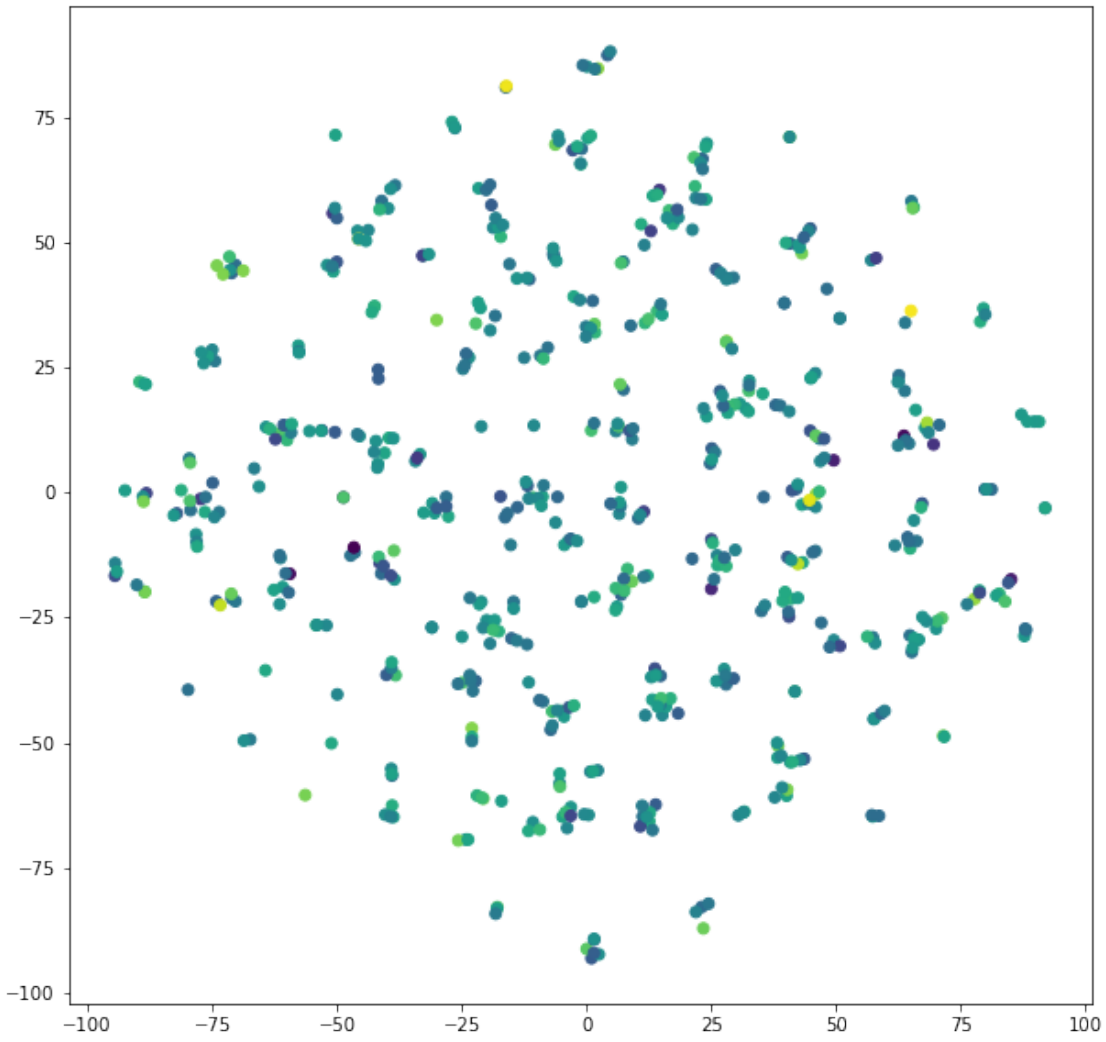
[130]:
```
# Plot TSNE scatter plot
plt.figure(figsize=(10,10))
plt.scatter(X_emb[:, 0], X_emb[:, 1], c=c_chol)
```

[130]:
```
<matplotlib.collections.PathCollection at 0x7f804c855ac0>
```



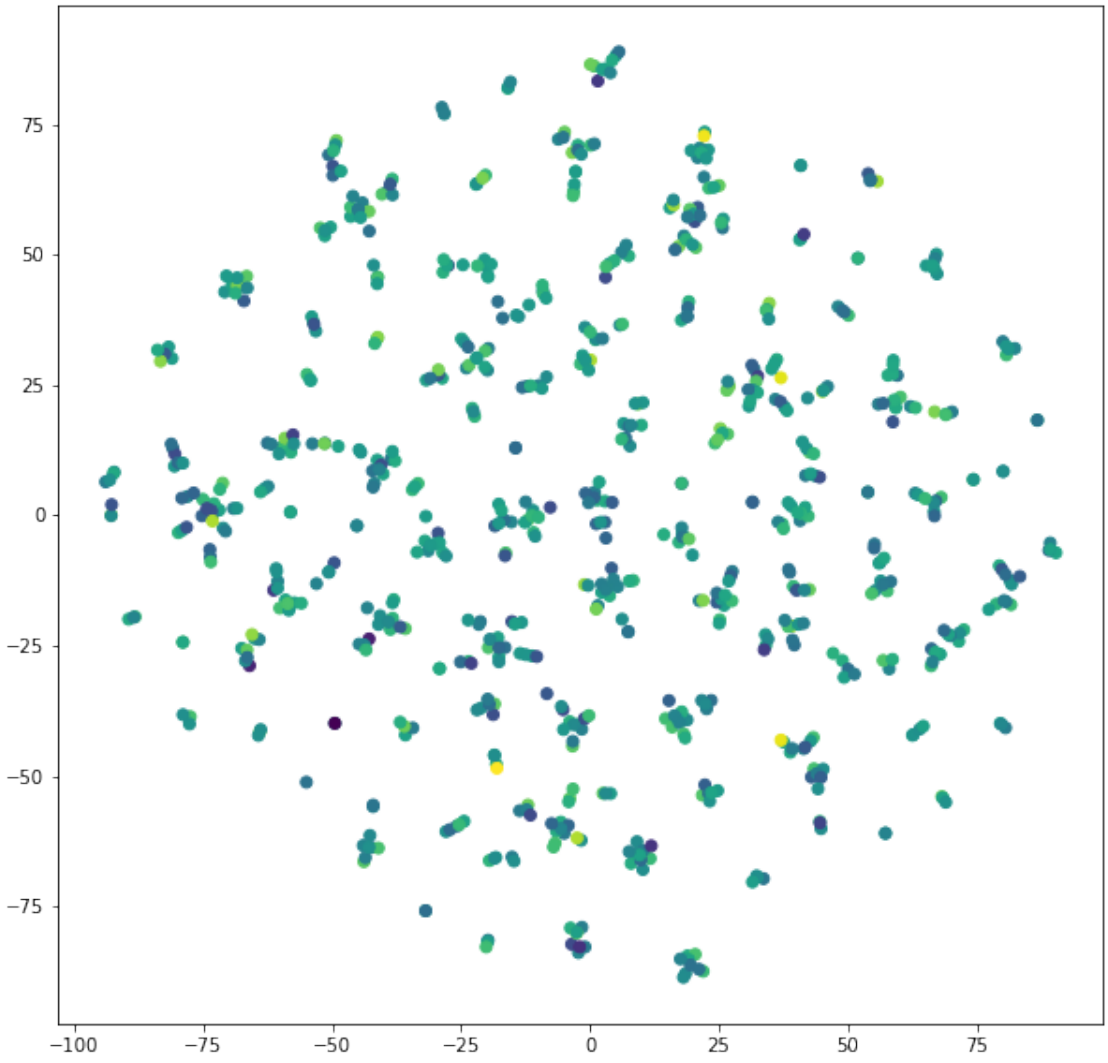
[143]:
```
# Plot TSNE scatter plot
plt.figure(figsize=(10,10))
plt.scatter(X_emb[:, 0], X_emb[:, 1], c=gmv)
```

[143]: <matplotlib.collections.PathCollection at 0x7f804da9c3a0>



[132]:
```
# Plot TSNE scatter plot
plt.figure(figsize=(10,10))
plt.scatter(X_emb[:, 0], X_emb[:, 1], c=c_cog)
```

[132]: <matplotlib.collections.PathCollection at 0x7f8048925a00>


75
50
25
0
−25
−50
−75
−100
−75
−50
−25
0
25
50
75

[ ]: