Elucidata Assignment - Pancreatic Cancer Analysis

Task 1 :

- What does the analysis say about the general behaviour of the different samples?
- Are the neuroendocrine tumors clearly separable from the adenocarcinoma tumors?
- What can be said about the variance of the PCA?

Task 2:

- Can you characterize the presence of IFN signature in pancreatic adenocarcinoma tumorsby assigning a score to each sample which denotes the positive or negative presence of IFN genes in the sample?
- How is the distribution of this score among the different samples?
- Based on this distribution can we identify the presence of high and low IFN subtypes in PAAD?

Import Some Required `Libraries`

In [1]:
```
from cmapPy.pandasGEXpress.parse import parse
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
from GSVA import gsva
# Some extras to look at the high dimensional data
from plotnine import *
from sklearn.manifold import TSNE
```

Load data using `cmappy`

In [2]:
```
PAAD_multi_data = parse("PAAD.gct", convert_neg_666 = True,make_multiindex = True)
PAAD_multi_data.multi_index_df
```

Out[2]:

participant_id	aab1	aab4	aab6	aab8	
sample_type	Primary solid Tumor	Primary solid Tumor	Primary solid Tumor	Primary solid Tumor	
mRNAseq_cluster	1.0	2.0	3.0	1.0	
bcr_patient_barcode	tcga-2j-aab1	tcga-2j-aab4	tcga-2j-aab6	tcga-2j-aab8	
bcr_patient_uuid	75119d1a-93e5-4ae7-9d60-69ee929a0772	33833131-1482-42d5-9cf5-01cade540234	70797499-16e6-48cc-8ae4-1e692713dad3	2e8f90f4-aed3-43b0-985c-dfdc2581f24f	a5 0
vital_status	dead	alive	dead	alive	
days_to_death	66.0	NaN	293.0	NaN	
days_to_last_followup	NaN	729.0	NaN	80.0	
additional_studies	NaN	NaN	NaN	NaN	
adenocarcinoma_invasion	yes	yes	yes	yes	

In [3]:
```
meta_data = parse("PAAD.gct", convert_neg_666 = True)
meta_data
```

Out[3]: `<cmapPy.pandasGEXpress.GCToo.GCToo at 0x2579297d128>`

Column Meta Data

In [4]:
```
1 meta_data.col_metadata_df
```

Out[4]:

chd cid	participant_id	sample_type	mRNAseq_cluster	bcr_patient_barcode	bcr_patient_uuid	vital_status	days_to_death	days_to_last_followup
aab1-Primary solid Tumor	aab1	Primary solid Tumor	1.0	tcga-2j-aab1	75119d1a-93e5-4ae7-9d60-69ee929a0772	dead	66.0	NaN
aab4-Primary solid Tumor	aab4	Primary solid Tumor	2.0	tcga-2j-aab4	33833131-1482-42d5-9cf5-01cade540234	alive	NaN	729.0
aab6-Primary solid Tumor	aab6	Primary solid Tumor	3.0	tcga-2j-aab6	70797499-16e6-48cc-8ae4-1e692713dad3	dead	293.0	NaN
aab8-Primary solid	aab8	Primary solid Tumor	1.0	tcga-2j-aab8	2e8f90f4-aed3-43b0-985c-dfdc2581f24f	alive	NaN	80.0

`Expression Data` : Samples of approx 20000 genes and 183 pancreatic cancer tumors

In [5]:
```
1 meta_data.data_df
```

Out[5]:

cid rid	aab1-Primary solid Tumor	aab4-Primary solid Tumor	aab6-Primary solid Tumor	aab8-Primary solid Tumor	aab9-Primary solid Tumor	aaba-Primary solid Tumor	aabe-Primary solid Tumor	aabf-Primary solid Tumor	aabh-Primary solid Tumor	aabi-Primary solid Tumor	...	aauh-Primary solid Tumor	aaui-Primary solid Tumor	aaul-Primary solid Tumor	a8t3-Primary solid Tumor	F
SLC35E2	7.45	8.1	7.2	8.0	7.65	8.1	8.2	8.2	7.55	8.45	...	8.45	7.95	8.3	8.05	
A1BG	6.40	5.8	6.4	5.8	6.70	6.6	6.3	6.5	5.70	6.30	...	7.10	7.10	6.7	7.00	
A1CF	4.70	5.7	3.0	5.1	4.40	4.2	1.6	6.8	6.00	NaN	...	5.40	6.40	6.5	4.40	
A2BP1	-1.00	1.1	NaN	NaN	0.10	NaN	NaN	1.7	0.40	-1.50	...	3.50	1.30	-0.3	NaN	
A2LD1	7.50	6.8	7.3	7.5	7.40	6.6	7.1	6.8	8.00	5.80	...	6.50	7.30	6.1	6.70	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
ZYG11B	9.20	9.3	9.4	9.4	9.30	9.9	9.1	9.5	8.90	8.30	...	9.70	9.20	9.5	9.50	
ZYX	12.90	12.4	13.5	12.5	13.00	12.2	12.9	12.6	12.70	12.50	...	12.40	12.60	13.5	12.50	

Check For `Null values` in Data

In [6]:
```
1 # For Column Meta Data
2 meta_data.col_metadata_df.isnull().sum()
```

Out[6]:
```
chd
participant_id                          0
sample_type                             0
mRNAseq_cluster                         5
bcr_patient_barcode                     0
bcr_patient_uuid                        0
                                     ...
withdrawn                               0
year_of_dcc_upload                      0
year_of_form_completion                 0
year_of_initial_pathologic_diagnosis    1
year_of_tobacco_smoking_onset         135
Length: 124, dtype: int64
```

In [7]:
```
1 # For expression Data
2 meta_data.data_df.isnull().sum()
```

Out[7]:
```
cid
aab1-Primary solid Tumor     645
aab4-Primary solid Tumor     532
aab6-Primary solid Tumor     983
aab8-Primary solid Tumor    1014
aab9-Primary solid Tumor     961
                            ...
a89d-Solid Tissue Normal     581
a89d-Primary solid Tumor     593
a8sy-Primary solid Tumor     829
a8lh-Primary solid Tumor     699
aapl-Primary solid Tumor     934
Length: 183, dtype: int64
```

OBSERVATIONS:

- Many Features have null values in both expression data and column meta data

Correlation Matrix for `column meta data`

In [9]:
```
import pandas as pd
import numpy as np

corr = meta_data.col_metadata_df.corr()
corr.style.background_gradient(cmap='coolwarm')
# 'RdBu_r' & 'BrBG' are other good diverging colormaps
```

Out[9]:

chd	mRNAseq_cluster	days_to_death	days_to_last_followup	additional_studies	age_at_initial_pathologic
chd					
mRNAseq_cluster	1.000000	0.195662	-0.248586	nan	
days_to_death	0.195662	1.000000	nan	nan	
days_to_last_followup	-0.248586	nan	1.000000	nan	
additional_studies	nan	nan	nan	nan	
age_at_initial_pathologic_diagnosis	-0.053023	-0.057989	-0.271381	nan	
amount_of_alcohol_consumption_per_day	-0.224789	-0.330821	0.016455	nan	
b_symptoms	nan	nan	nan	nan	
clinical_m	nan	nan	nan	nan	
clinical_n	nan	nan	nan	nan	
clinical_stage	nan	nan	nan	nan	

Corelation Matrix for `expression data`

In [10]:
```
import pandas as pd
import numpy as np

corr = meta_data.data_df.corr()
corr.style.background_gradient(cmap='coolwarm')
# 'RdBu_r' & 'BrBG' are other good diverging colormaps
```

Out[10]:

cid	aab1-Primary solid Tumor	aab4-Primary solid Tumor	aab6-Primary solid Tumor	aab8-Primary solid Tumor	aab9-Primary solid Tumor	aaba-Primary solid Tumor	aabe-Primary solid Tumor	aabf-Primary solid Tumor	aabh-Primary solid Tumor	aabi-Primary solid Tumor	aabk-Primary solid Tumor	aabo-Primary solid Tumor	aabp-Primary solid Tumor	I
cid														
aab1-Primary solid Tumor	1.000000	0.947819	0.894680	0.924493	0.934009	0.928397	0.943618	0.949707	0.947124	0.861580	0.936252	0.935989	0.804305	0
aab4-Primary solid Tumor	0.947819	1.000000	0.909971	0.911957	0.940745	0.906942	0.941612	0.959046	0.957116	0.859409	0.951279	0.932207	0.799655	0
aab6-Primary solid Tumor	0.894680	0.909971	1.000000	0.900083	0.903004	0.885412	0.912631	0.911457	0.906180	0.869485	0.888683	0.914525	0.842576	0

Checking for `duplicate genes` in gene `expression data`

In [11]:
```
print("Number of duplicate genes in gene expression data",meta_data.data_df.T.shape[1] - meta_data.data_df.T.columns
```

```
Number of duplicate genes in gene expression data 0
```

Checking for `duplicate genes` in gene `column meta data`

In [12]:
```
print("Number of duplicate in column meta data ",meta_data.col_metadata_df.T.shape[1] - meta_data.col_metadata_df.T.
```

```
Number of duplicate in column meta data  0
```

OBSERVATIONS

- Some features have `all Null values` so we simply `drop them` and others which have `some Null values` we will `impute them` .
- Some features have only one value and some are id which we do not need for visualization purpose. As features have same values have `zero variance` .
- There is `no duplicates` of genes in gene expression data

In [13]:
```
meta_data.col_metadata_df.drop(columns = ["participant_id","sample_type","bcr_patient_barcode","bcr_patient_uuid","v
meta_data.col_metadata_df
```

Out[13]:

chd cid	mRNAseq_cluster	days_to_death	days_to_last_followup	adenocarcinoma_invasion	age_at_initial_pathologic_diagnosis	alcohol_history_d
aab1-Primary solid Tumor	1.0	66.0	NaN	yes	65	
aab4-Primary solid Tumor	2.0	NaN	729.0	yes	48	
aab6-Primary solid Tumor	3.0	293.0	NaN	yes	75	
aab8-Primary solid	1.0	NaN	80.0	yes	71	

Missing value Imputation

- Some data contain most of the `NULL values` . Features having many NULL values have `low variance` and they are not good to use so we simply drop them. We set a `thresold value` lets say about `90 %` , features with more than 90 % of values are NULL is been dropped. So basically our data have 183 rows in column meta data so we `drop features > 170 NULL values` .
- And in Expression data we `remove features which > 2000 Null Values` .

OBSERVATIONS:

- `column_meta_data` have many `categorical features` so we `impute missing values` in them using `most frequent valueS.`
- `Expression data` have `float values` so we simply `impute missing values` in them using `mean` .

In [14]:
```
a = (( meta_data.col_metadata_df.isnull().sum() > 170).astype(np.int64).sum())
print("Number of features have more than 170 null values in column_meta_data is : ",a)
```

```
Number of features have more than 170 null values in column_meta_data is :  7
```

In [15]:
```
a = (( meta_data.data_df.isnull().sum() > 2000).astype(np.int64).sum())
print("Number of features have more than 10K null values in gene_data is : ",a)
```

```
Number of features have more than 10K null values in gene_data is :  0
```

In [16]:
```
meta_data.col_metadata_df.drop(columns = ["project_code","informed_consent_verified","disease_code","anatomic_neopla
meta_data.col_metadata_df
```

Out[16]:

chd cid	mRNAseq_cluster	days_to_death	days_to_last_followup	adenocarcinoma_invasion	age_at_initial_pathologic_diagnosis	alcohol_history_d
aab1-Primary solid Tumor	1.0	66.0	NaN	yes	65	
aab4-Primary solid Tumor	2.0	NaN	729.0	yes	48	
aab6-Primary solid Tumor	3.0	293.0	NaN	yes	75	
aab8-Primary solid	1.0	NaN	80.0	yes	71	

In [17]:
```
column_meta_data_imputed = meta_data.col_metadata_df.fillna(meta_data.col_metadata_df.mode().iloc[0])
```

In [18]:
```
row_meta_data_imputed = meta_data.data_df.fillna(meta_data.data_df.mean().iloc[0])
```

Check `Correlation Matrix` Again If we have improved

In [19]:
```
import pandas as pd
import numpy as np

corr = column_meta_data_imputed.corr()
corr.style.background_gradient(cmap='coolwarm')
# 'RdBu_r' & 'BrBG' are other good diverging colormaps
```

Out[19]:

chd	mRNAseq_cluster	days_to_death	days_to_last_followup	age_at_initial_pathologic_diagnosis	amount_of_alcc
chd					
mRNAseq_cluster	1.000000	0.078207	-0.129234	-0.051935	
days_to_death	0.078207	1.000000	-0.216437	0.030414	
days_to_last_followup	-0.129234	-0.216437	1.000000	-0.212643	
age_at_initial_pathologic_diagnosis	-0.051935	0.030414	-0.212643	1.000000	
amount_of_alcohol_consumption_per_day	-0.121034	-0.068933	0.019152	0.100952	
day_of_form_completion	-0.047042	-0.190378	0.245902	0.049469	
days_to_birth	0.050728	-0.031978	0.211882	-0.999659	
frequency_of_alcohol_consumption	-0.060112	-0.004294	0.008066	-0.065731	
icd_o_3_histology	0.130562	0.132266	-0.220854	-0.017963	
lymph_node_examined_count	0.107049	-0.010304	0.026858	-0.106486	
maximum_tumor_dimension	-0.132414	-0.023340	0.040089	0.052797	
month_of_form_completion	0.021893	0.042028	0.208117	-0.146607	
number_of_lymphnodes_positive_by_he	0.130176	-0.010149	-0.098343	-0.020416	
number_pack_years_smoked	-0.033890	0.004935	0.076007	-0.049502	
stopped_smoking_year	0.025837	-0.020493	0.028402	-0.212330	
system_version	-0.069668	-0.209828	-0.393350	0.162191	
tobacco_smoking_history	0.053904	0.094265	0.041434	-0.048662	
year_of_form_completion	-0.109751	-0.123767	0.211108	0.012081	
year_of_initial_pathologic_diagnosis	0.047872	-0.491289	-0.295380	0.100153	
year_of_tobacco_smoking_onset	0.096264	-0.020636	0.011404	-0.302539	

In [20]:
```
corr = row_meta_data_imputed.corr()
corr.style.background_gradient(cmap='coolwarm')
# 'RdBu_r' & 'BrBG' are other good diverging colormaps
```

Out[20]:

cid	aab1-Primary solid Tumor	aab4-Primary solid Tumor	aab6-Primary solid Tumor	aab8-Primary solid Tumor	aab9-Primary solid Tumor	aaba-Primary solid Tumor	aabe-Primary solid Tumor	aabf-Primary solid Tumor	aabh-Primary solid Tumor	aabi-Primary solid Tumor	aabk-Primary solid Tumor	aabo-Primary solid Tumor	aabp-Primary solid Tumor	I
cid														
aab1-Primary solid Tumor	1.000000	0.861550	0.791152	0.815231	0.834241	0.832841	0.848138	0.863252	0.855566	0.772314	0.847753	0.841883	0.710505	0
aab4-Primary solid Tumor	0.861550	1.000000	0.802784	0.808989	0.840707	0.814287	0.847479	0.881836	0.875578	0.771173	0.874801	0.841262	0.705568	0
aab6-Primary solid Tumor	0.791152	0.802784	1.000000	0.791320	0.802205	0.779286	0.806550	0.798507	0.800665	0.769185	0.786878	0.816756	0.745863	0

OBSERVATION:

- Corelation improved there are no black cells.

STEPS FOR `TASK-1`

Apply Scaling before perform PCA due to different Scales of Features

- Our expression data have some `negative values` so we are calculate `means` of every genes samples and `replace` these negative values with `mean` of that feature
- Also some of `values` in expression data is `0` so we are `replace` it also with `mean` value
- Then apllying `log2 scaling` and `quantile normalization` on data.

We want to stratify these tumor samples by the type of pancreatic cancer they exhibit. For this, apply dimensionality reduction techniques (PCA) to find these two groups within this multi-dimensional data.

- `Visualize` the data whole data using PCA.
- Write `observations` for each plot.
- Explaining `Varience` in PCA
- Remove the neuroendocrine tumors from the dataset so that it contains only the adenocarcinoma tumor samples. The histology for the different tumor samples is contained in the GCT file.

In [22]:
```
row_meta_data_imputed = row_meta_data_imputed.T
row_meta_data_imputed
```

Out[22]:

rid cid	SLC35E2	A1BG	A1CF	A2BP1	A2LD1	A2ML1	A2M	A4GALT	A4GNT	AAA1	...	ZWINT	ZXDA	ZXDB	ZXDC	ZYG11A	Z
aab1-Primary solid Tumor	7.45	6.4	4.7	-1.000000	7.5	6.400000	14.3	10.6	8.8	1.000000	...	8.6	6.2	9.0	9.9	7.600212	
aab4-Primary solid Tumor	8.10	5.8	5.7	1.100000	6.8	7.600212	14.0	10.2	5.6	-1.200000	...	8.8	5.8	8.5	10.0	7.600212	
aab6-Primary solid Tumor	7.20	6.4	3.0	7.600212	7.3	10.800000	13.1	10.1	0.2	0.200000	...	9.1	3.9	8.1	10.0	-0.800000	
aab8-Primary solid	8.00	5.8	5.1	7.600212	7.5	4.100000	13.8	8.6	3.2	-0.100000	...	8.9	5.2	8.5	9.7	1.900000	

In [23]:
```
for c in row_meta_data_imputed.columns:
   for j in range(row_meta_data_imputed.shape[0]):
       if(row_meta_data_imputed[c][j] <= 0 and row_meta_data_imputed[c][j] < row_meta_data_imputed[c].mean()):
           row_meta_data_imputed[c][j] = row_meta_data_imputed[c].mean()
```

In [24]:
```
row_meta_data_logScale = np.log2(row_meta_data_imputed)
row_meta_data_logScale
```

Out[24]:

rid cid	SLC35E2	A1BG	A1CF	A2BP1	A2LD1	A2ML1	A2M	A4GALT	A4GNT	AAA1	...	ZWINT	ZXDA	ZXDB
aab1-Primary solid Tumor	2.897240	2.678072	2.232661	1.059873	2.906891	2.678072	3.837943	3.405993	3.137504	0.000000	...	3.104337	2.632268	3.169925
aab4-Primary solid Tumor	3.017922	2.536053	2.510962	0.137504	2.765535	2.926040	3.807355	3.350497	2.485427	1.559010	...	3.137504	2.536053	3.087463
aab6-Primary solid Tumor	2.847997	2.678072	1.584962	2.926040	2.867897	3.432960	3.711495	3.336283	-2.321928	-2.321928	...	3.185867	1.963474	3.017922
aab8-Primary solid	3.000000	2.536053	2.350497	2.926040	2.906891	2.035624	3.786596	3.104337	1.678072	1.570062	...	3.153805	2.378511	3.087463

In [25]:
```
from sklearn.preprocessing import quantile_transform

row_meta_data_quantile = quantile_transform(row_meta_data_logScale, n_quantiles=10, random_state=0, copy=True)
row_meta_data_quantile
```

Out[25]:
```
array([[0.20330299, 0.51909895, 0.39446749, ..., 0.44444444, 0.55555556,
        0.63885706],
       [0.59282102, 0.26451572, 0.61935892, ..., 0.27806209, 0.22222222,
        0.31386939],
       [0.10998002, 0.51909895, 0.16545217, ..., 0.27806209, 0.44444444,
        0.36954742],
       ...,
       [0.16508077, 0.90456325, 0.05555556, ..., 0.27806209, 0.10160843,
        0.76091463],
       [0.10998002, 0.34954709, 0.44444444, ..., 0.11111111, 0.08226672,
        0.07273927],
       [0.93311885, 0.7042389 , 0.09649809, ..., 0.89944703, 0.66666667,
        0.61053734]])
```

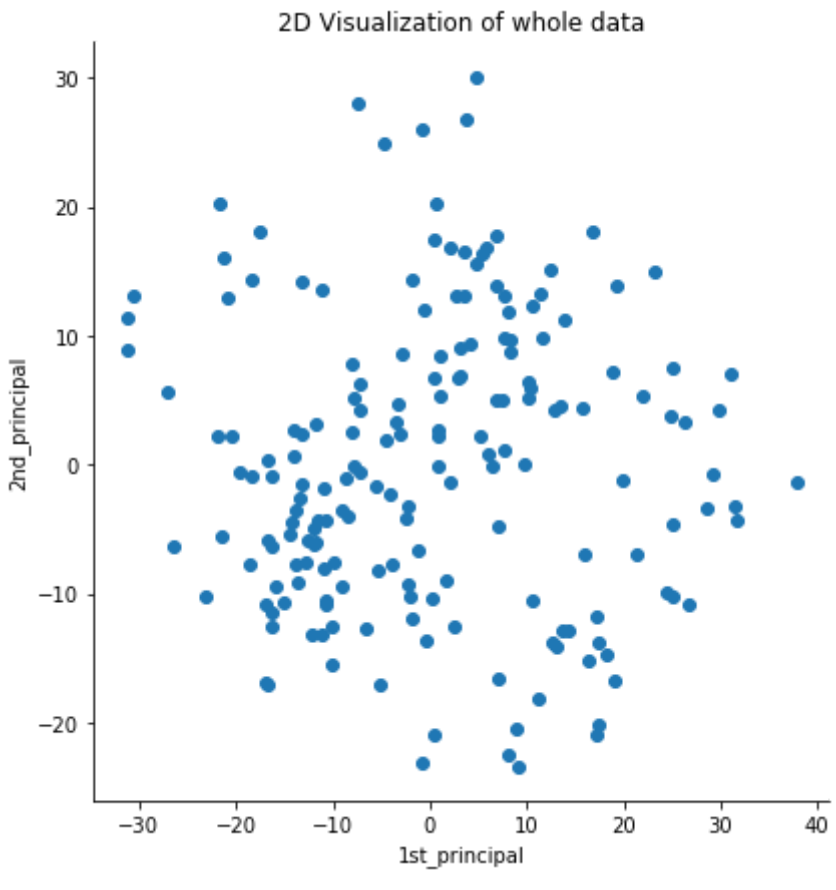
Visualize the expression data using PCA.

In [23]:
```
# initializing the pca
from sklearn.decomposition import PCA

# configuring the parameteres
# the number of components = 2
pcamodel = PCA(n_components=2)
pca = pcamodel.fit_transform(row_meta_data_quantile)

# pca_reduced will contain the 2-d projects of simple data
print("shape of pca_reduced.shape = ", pca.shape)
```

```
shape of pca_reduced.shape =  (183, 2)
```

Visualize Whole Data

In [24]:
```
# attaching the label for each 2-d data point

pca_data = np.vstack((pca.T)).T

# creating a new data fram which help us in ploting the result data
pca_df = pd.DataFrame(data=pca_data, columns=("1st_principal", "2nd_principal"))

sns.FacetGrid(pca_df, height=6).map(plt.scatter, '1st_principal', '2nd_principal')
plt.title("2D Visualization of whole data")
plt.show()
print("Shape OF Data after reducing components from 18465 to 2 : ",pca_df.shape)
pca_df.head()
```




```
Shape OF Data after reducing components from 18465 to 2 :  (183, 2)
```

Out[24]:

	1st_principal	2nd_principal
0	-10.896782	-1.765180
1	-4.133268	-2.222903
2	-11.495398	-4.291451
3	-3.988638	-7.678968
4	0.338806	6.675309

Let's look on variance explained by 2 principal components of whole data

In [25]:
```
#The amount of variance that each PC explains
var= pcamodel.explained_variance_ratio_
print("Variance Explained by 1st_principal and 2nd_principal is : ",var)
```

```
Variance Explained by 1st_principal and 2nd_principal is :  [0.13746665 0.08399517]
```

In [26]:
```
#Cumulative Variance explains
var1=np.cumsum(np.round(pcamodel.explained_variance_ratio_, decimals=4)*100)

print ("Cumulative Variance Explained by 1st_principal and 2nd_principal is : ",var1b
```

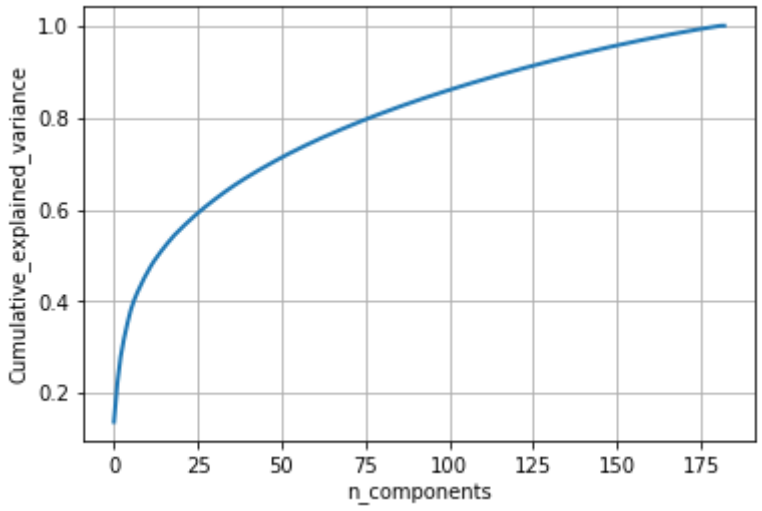
```
Cumulative Variance Explained by 1st_principal and 2nd_principal is :  [13.75 22.15]
```

OBSERVATION:

- 22.15% variance explained by 2 principal components

Check For variance explained by all 183 components

In [27]:
```
pcamodel = PCA(n_components=183)
pca = pcamodel.fit_transform(row_meta_data_quantile)

percentage_var_explained = pcamodel.explained_variance_ / np.sum(pcamodel.explained_variance_);

cum_var_explained = np.cumsum(percentage_var_explained)

# Plot the PCA spectrum
plt.figure(1, figsize=(6, 4))

plt.clf()
plt.plot(cum_var_explained, linewidth=2)
plt.axis('tight')
plt.grid()
plt.xlabel('n_components')
plt.ylabel('Cumulative_explained_variance')
plt.show()
```



OBSERVATION :

- As we can see from above plot almost 90 % variance is explained by almost 125 components

Visualize Data using different Samples

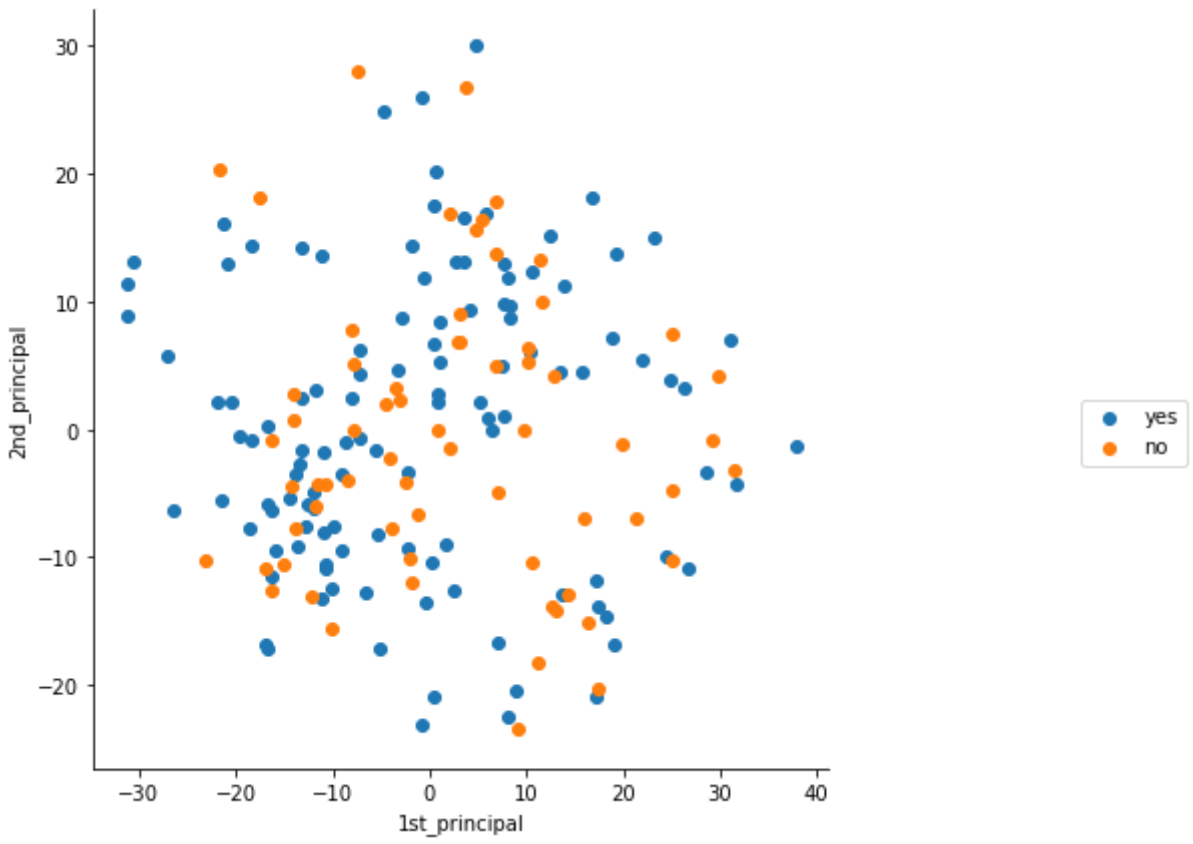
In [28]:
```
# initializing the pca
from sklearn.decomposition import PCA

# configuring the parameteres
# the number of components = 2
pcamodel = PCA(n_components=2)
pca = pcamodel.fit_transform(row_meta_data_quantile)

# pca_reduced will contain the 2-d projects of simple data
print("shape of pca_reduced.shape = ", pca.shape)
```

```
shape of pca_reduced.shape =  (183, 2)
```

Let see what analysis say about alcohol_history_documented feature

In [29]:
```
# attaching the label for each 2-d data point
Y = column_meta_data_imputed["alcohol_history_documented"]
pca_data = np.vstack((pca.T, Y)).T
print(pca_data.shape)
# creating a new data fram which help us in ploting the result data
pca_df = pd.DataFrame(data=pca_data, columns=("1st_principal", "2nd_principal", "label"))
sns.FacetGrid(pca_df, hue="label", height=6).map(plt.scatter, '1st_principal', '2nd_principal')
plt.legend(loc='best',bbox_to_anchor=(1, 0., 0.5, 0.5))
plt.show()
```

```
(183, 3)
```



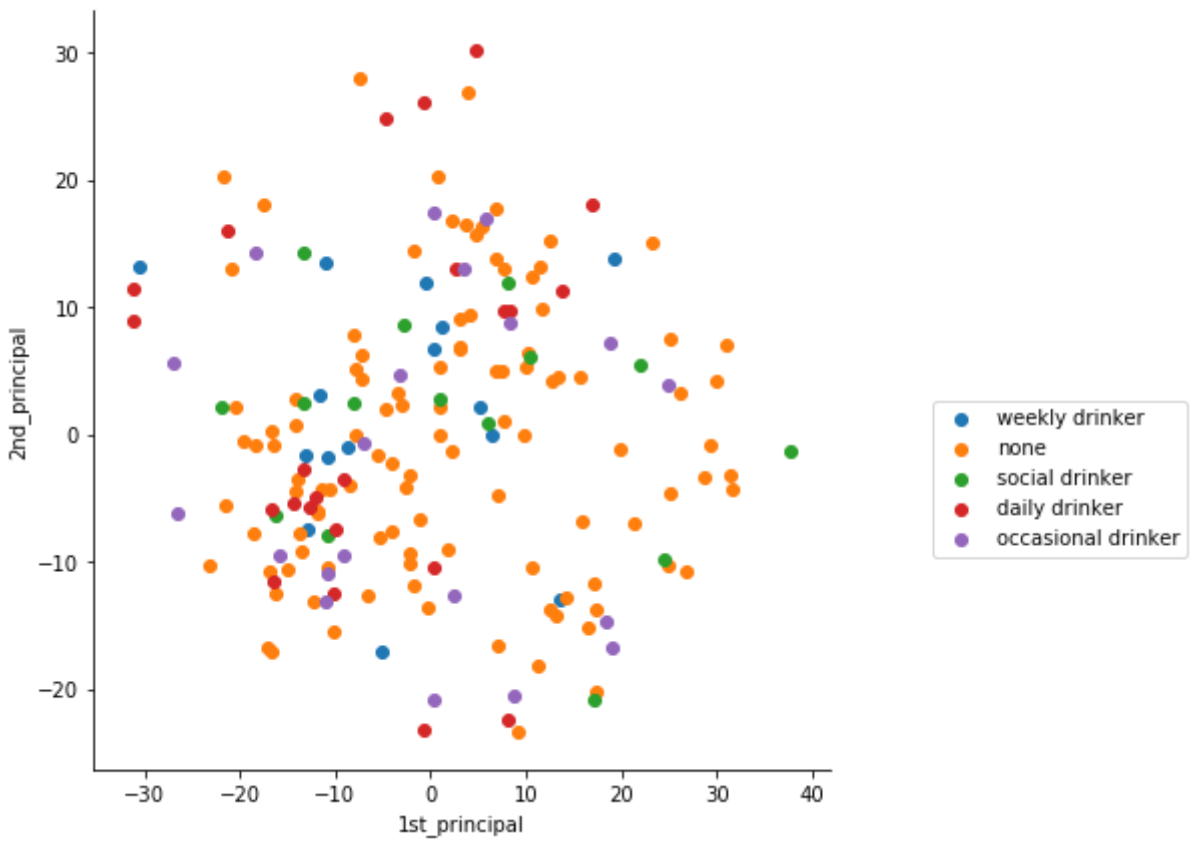
OBSERVATION

- More people have `alcoholic history`

For `alcoholic_exposure_category`

In [30]:
```
# attaching the label for each 2-d data point
Y = column_meta_data_imputed["alcoholic_exposure_category"]
pca_data = np.vstack((pca.T, Y)).T
print(pca_data.shape)
# creating a new data fram which help us in ploting the result data
pca_df = pd.DataFrame(data=pca_data, columns=("1st_principal", "2nd_principal", "label"))
sns.FacetGrid(pca_df, hue="label", height=6).map(plt.scatter, '1st_principal', '2nd_principal')
plt.legend(loc='best',bbox_to_anchor=(1, 0., 0.5, 0.5))
plt.show()
```

```
(183, 3)
```



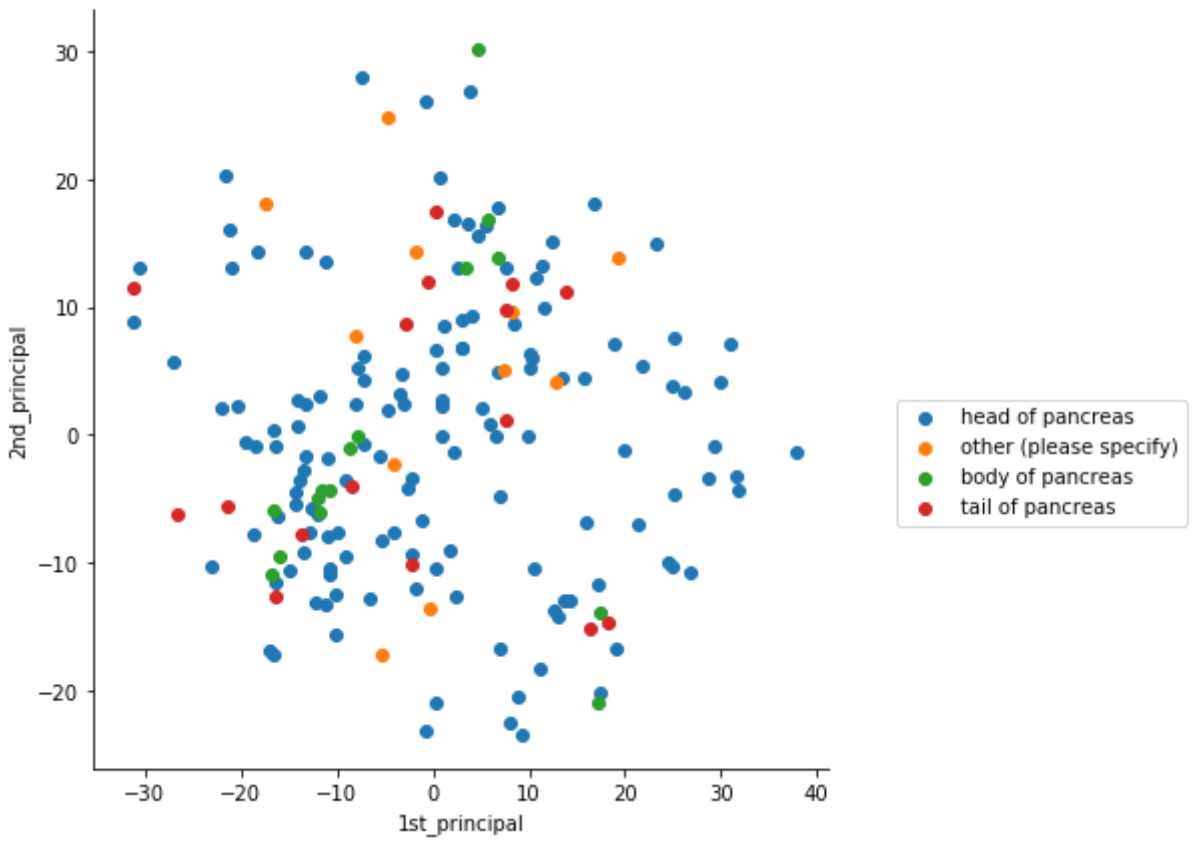
OBSERVATION:

- Many people comes under `none` category.
- Many people do not consume alcohol on daily basis

For `anatomic_neoplasm_subdivision`

In [31]:
```
# attaching the label for each 2-d data point
Y = column_meta_data_imputed["anatomic_neoplasm_subdivision"]
pca_data = np.vstack((pca.T, Y)).T
print(pca_data.shape)
# creating a new data fram which help us in ploting the result data
pca_df = pd.DataFrame(data=pca_data, columns=("1st_principal", "2nd_principal", "label"))
sns.FacetGrid(pca_df, hue="label", height=6).map(plt.scatter, '1st_principal', '2nd_principal')
plt.legend(loc='best',bbox_to_anchor=(1, 0., 0.5, 0.5))
plt.show()
```

```
(183, 3)
```



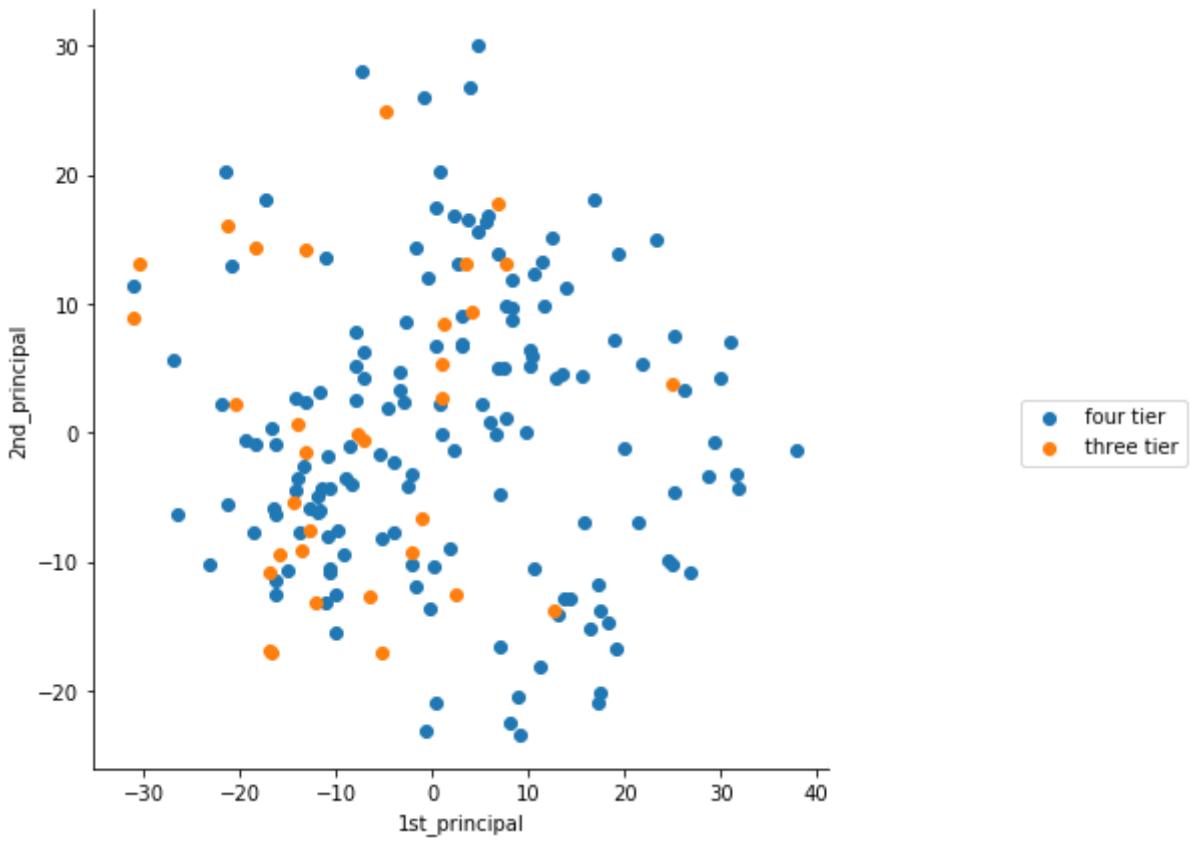
OBSERVATION

- Many Pancreatic Cancer is on `Head of Pancreas`

For `histologic_grading_tier_category`

In [32]:
```
# attaching the label for each 2-d data point
Y = column_meta_data_imputed["histologic_grading_tier_category"]
pca_data = np.vstack((pca.T, Y)).T
print(pca_data.shape)
# creating a new data fram which help us in ploting the result data
pca_df = pd.DataFrame(data=pca_data, columns=("1st_principal", "2nd_principal", "label"))
sns.FacetGrid(pca_df, hue="label", height=6).map(plt.scatter, '1st_principal', '2nd_principal')
plt.legend(loc='best',bbox_to_anchor=(1, 0., 0.5, 0.5))
plt.show()
```

```
(183, 3)
```



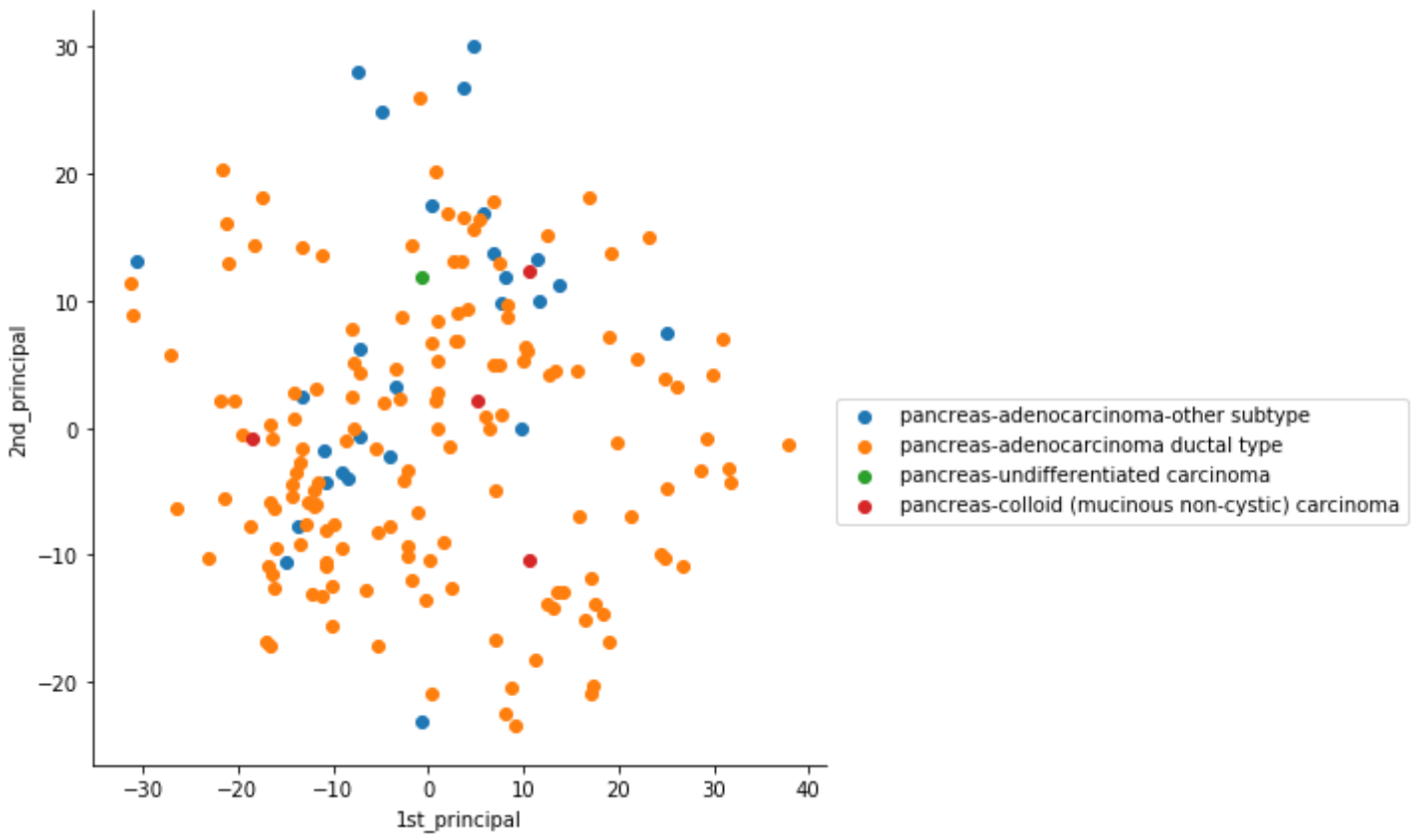
OBSERVATION

- Many are of `four tier`

For `histological_type`

In [33]:
```
# attaching the label for each 2-d data point
Y = column_meta_data_imputed["histological_type"]
pca_data = np.vstack((pca.T, Y)).T
print(pca_data.shape)
# creating a new data fram which help us in ploting the result data
pca_df = pd.DataFrame(data=pca_data, columns=("1st_principal", "2nd_principal", "label"))
sns.FacetGrid(pca_df, hue="label", height=6).map(plt.scatter, '1st_principal', '2nd_principal')
plt.legend(loc='best',bbox_to_anchor=(1, 0., 0.5, 0.5))
plt.show()
```

```
(183, 3)
```



OBSERVATION:

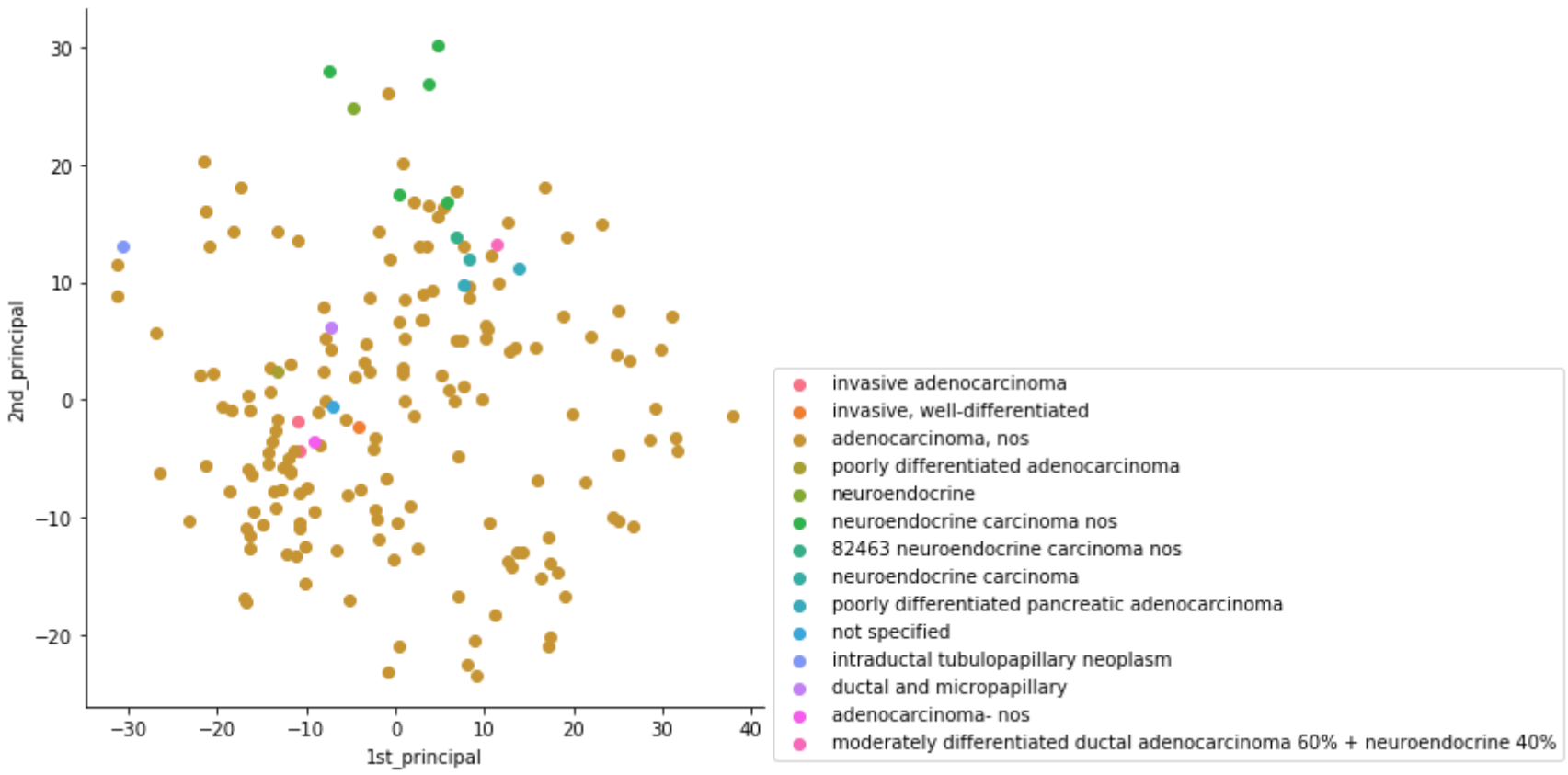
- Many tumors are pancreas adenocarcinoma `ductal type` which `most common type of exocrine tumors` .

For `histological_type_other`

- In histological_type_other feature `many values are NAN` so `impute them` using `most_frequent value` and plot PCA in 2D

In [34]:
```
# attaching the label for each 2-d data point
Y = column_meta_data_imputed["histological_type_other"]
pca_data = np.vstack((pca.T, Y)).T
print(pca_data.shape)
# creating a new data fram which help us in ploting the result data
pca_df = pd.DataFrame(data=pca_data, columns=("1st_principal", "2nd_principal", "label"))
sns.FacetGrid(pca_df, hue="label", height=6).map(plt.scatter, '1st_principal', '2nd_principal')
plt.legend(loc='best',bbox_to_anchor=(1, 0., 0.5, 0.5))
plt.show()
```

```
(183, 3)
```



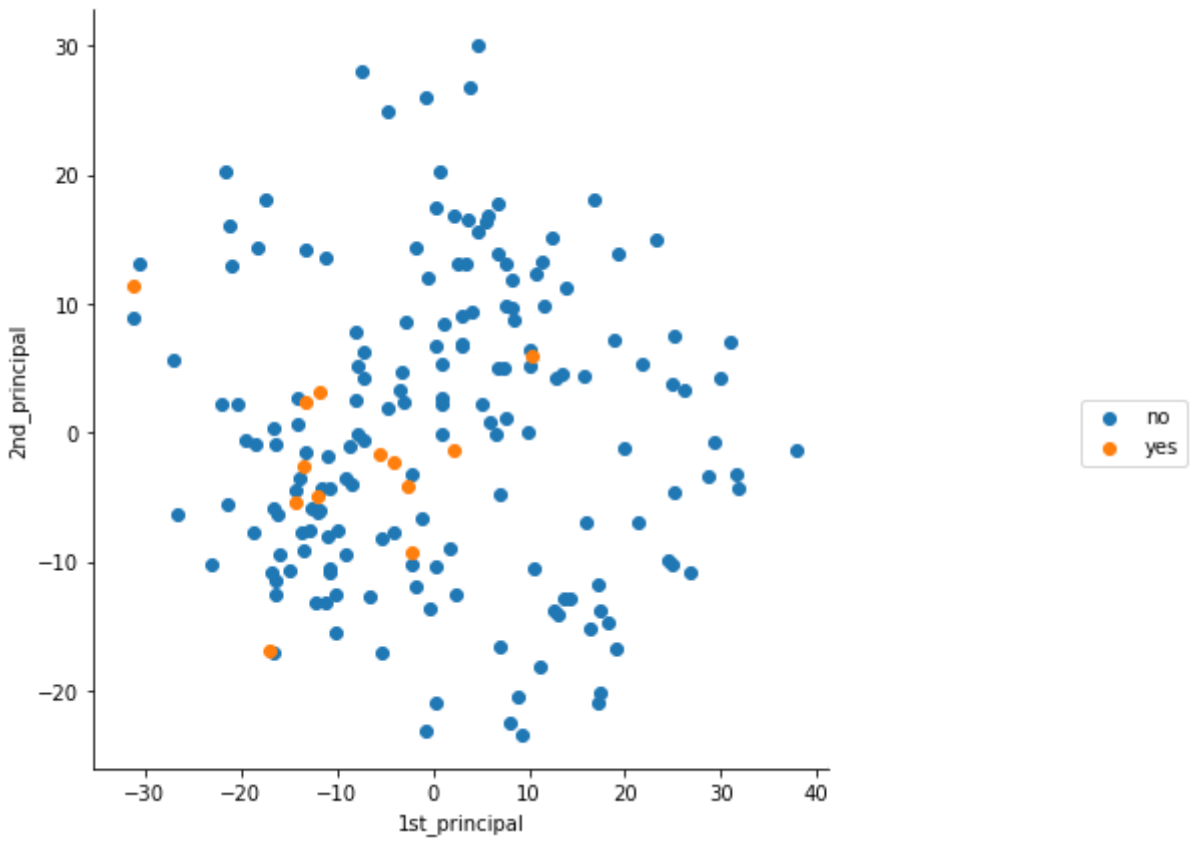
OBSERVATIONS:

- Many pancreatic cancer is of `adenocarcinoma type` which is also `most common type` of pancreatic cancer.
- Also `Neuroendocrine tumors and neuroendocrine carcinoma nos` are some `little seperable` from adenocarcinoma tumors

For `history_of_chronic_pancreatitis`

In [35]:
```
# attaching the label for each 2-d data point
Y = column_meta_data_imputed["history_of_chronic_pancreatitis"]
pca_data = np.vstack((pca.T, Y)).T
print(pca_data.shape)
# creating a new data fram which help us in ploting the result data
pca_df = pd.DataFrame(data=pca_data, columns=("1st_principal", "2nd_principal", "label"))
sns.FacetGrid(pca_df, hue="label", height=6).map(plt.scatter, '1st_principal', '2nd_principal')
plt.legend(loc='best',bbox_to_anchor=(1, 0., 0.5, 0.5))
plt.show()
```

```
(183, 3)
```



OBSERVATIONS:

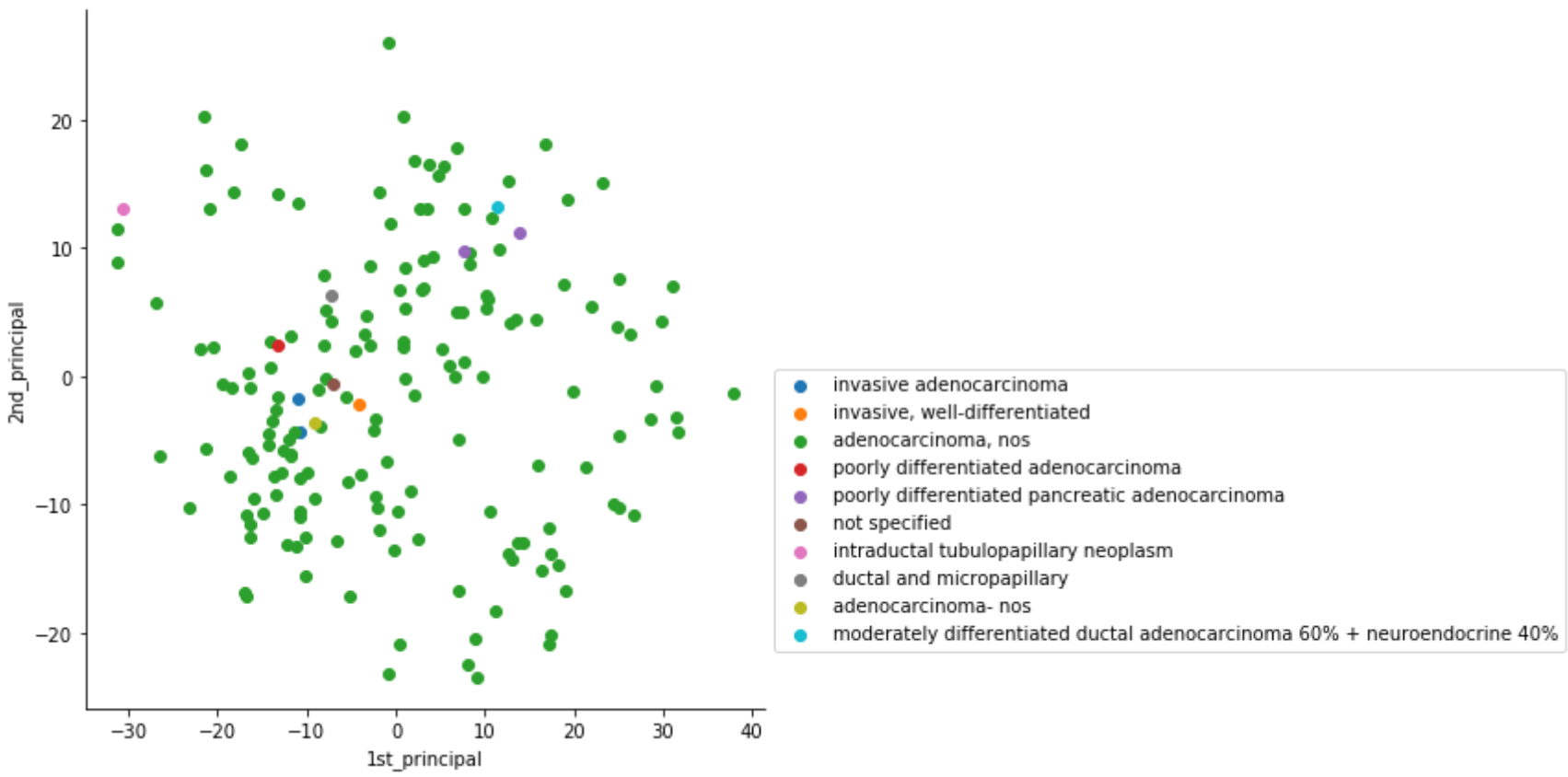
- Chronic pancreatitis is inflammation of the pancreas that does not heal or improve—it gets worse over time and leads to permanent damage.
- Many do not have history of Chronic pancreatitis

Remove the neuroendocrine tumors from the dataset so that it contains only the adenocarcinoma tumor samples and Visualize data.

In [38]:
```
column_meta_data_neuro_removed = column_meta_data_imputed.copy()

neuroendocrine = []
for val in column_meta_data_neuro_removed.histological_type_other:

    if(val == "neuroendocrine" or val == "neuroendocrine carcinoma nos" or val == "82463 neuroendocrine carcinoma nc

        neuroendocrine.append(np.NaN)

    else:
        neuroendocrine.append(val)
column_meta_data_neuro_removed.histological_type_other =   neuroendocrine

```

In [41]:
```
# attaching the label for each 2-d data point
Y = column_meta_data_neuro_removed.histological_type_other
pca_data = np.vstack((pca.T, Y)).T
print(pca_data.shape)
# creating a new data fram which help us in ploting the result data
pca_df = pd.DataFrame(data=pca_data, columns=("1st_principal", "2nd_principal", "label"))
sns.FacetGrid(pca_df, hue="label", height=6).map(plt.scatter, '1st_principal', '2nd_principal')
plt.legend(loc='best',bbox_to_anchor=(1, 0., 0.5, 0.5))
plt.show()
```

```
(183, 3)
```



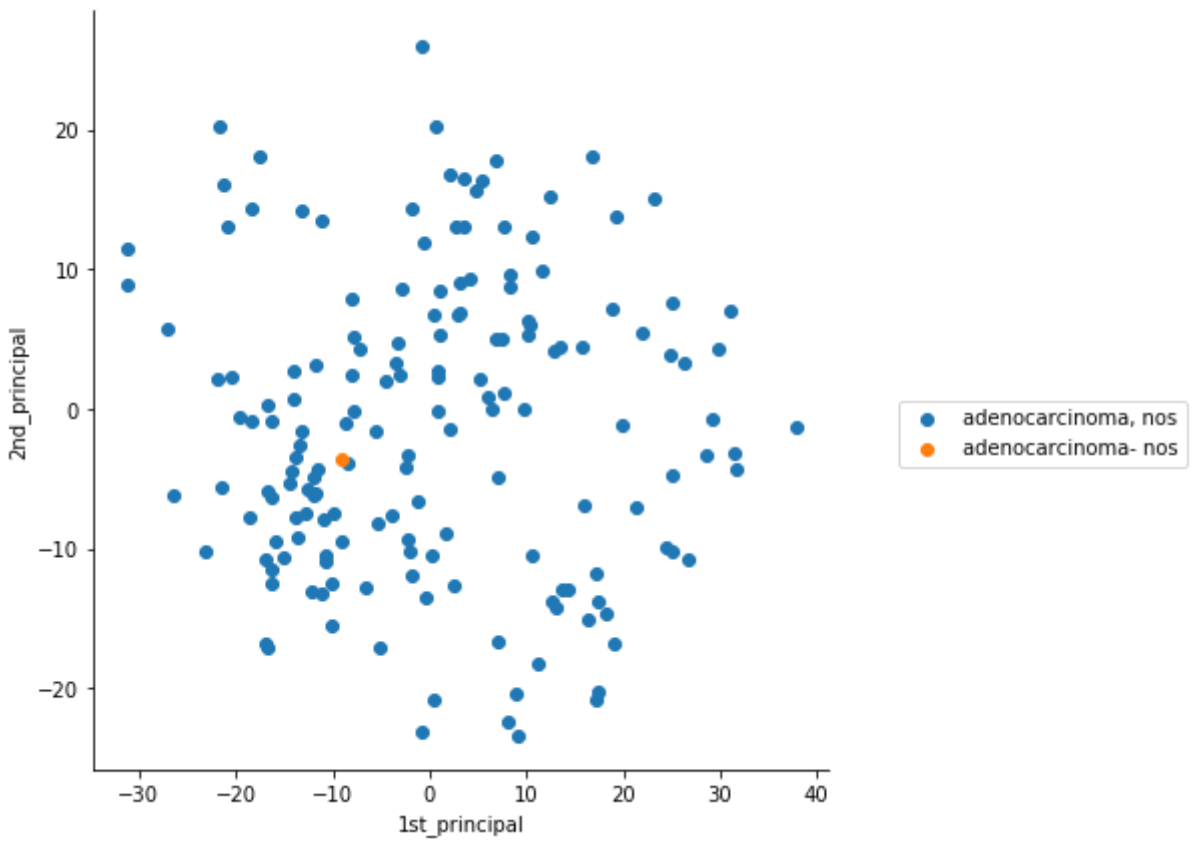
Now Keep only the `adenocarcinoma-nos` tumors from the dataset so that it contains only the `adenocarcinoma tumor samples` and Visualize data.

In [39]:
```
column_meta_data_adeno = column_meta_data_imputed.copy()

adenocarcinoma = []
for val in column_meta_data_adeno.histological_type_other:

    if(val == "neuroendocrine" or val == "invasive adenocarcinoma" or val == "poorly differentiated pancreatic adenc

        adenocarcinoma.append(np.NaN)

    else:
        adenocarcinoma.append(val)
column_meta_data_adeno.histological_type_other =   adenocarcinoma

```

For `histological_type_other`

In [43]:
```
# attaching the label for each 2-d data point
Y = column_meta_data_adeno.histological_type_other
pca_data = np.vstack((pca.T, Y)).T
print(pca_data.shape)
# creating a new data fram which help us in ploting the result data
pca_df = pd.DataFrame(data=pca_data, columns=("1st_principal", "2nd_principal", "label"))
sns.FacetGrid(pca_df, hue="label", height=6).map(plt.scatter, '1st_principal', '2nd_principal')
plt.legend(loc='best',bbox_to_anchor=(1, 0., 0.5, 0.5))
plt.show()
```

```
(183, 3)
```



OBSERVATIONS:

- After removing `Neuroendocrine tumors` we can see most of the type of tumors are of `adenocarcinoma tumor`

STEPS FOR TASK 2

- Running GSVA in Python: Run GSVA through the docker given for gsva python https://github.com/jason-weirather/GSVA (https://github.com/jason-weirather/GSVA), https://hub.docker.com/r/vacation/gsva (https://hub.docker.com/r/vacation/gsva)
- Visulising the : Plotting 25 genes (genes responsible for type 1 Interferons) in homo sapiens using gene expression data for pancreatic adenocarcinoma. So we are use two column meta data here on is histology_type and other is histology_type other for for pancreatic adenocarcinoma only
- Distribution of IFN genes

In [27]:
```
# Load all 25 genes

ifn_genes = pd.read_csv("type1_IFN.txt",sep="\t",header=None )
ifn_genes.columns = ["genes"]
ifn_genes
```

Out[27]:

	genes
0	IFIT1
1	IFI44
2	IFIT3
3	MX2
4	OAS1
5	OAS3
6	BST2
7	IFITM1
8	MX1
9	STAT1
10	IFI27
11	CXCL10
12	IFI16
13	IFI30
14	IFIH1
15	IFIT2
16	IFITM2
17	IRF1
18	IRF9
19	IRGM
20	ISG15
21	OAS2
22	PSME1
23	SOCS1
24	STAT2

Load `expression data` of these 25 genes and `plot PCA curves` for histology of tumors

In [29]:
```
ifn_gene_list = []
for val in ifn_genes.genes:

    ifn_gene_list.append(val)

print("Number of type 1 ifn genes",len(ifn_gene_list))

ifn_gene_df = row_meta_data_logScale[ifn_gene_list]
ifn_gene_df
```

```
Number of type 1 ifn genes 25
```

In [46]:
```
# Quantile transform the 25 gene data

from sklearn.preprocessing import quantile_transform

ifn_gene_data_quantile = quantile_transform(ifn_gene_df, n_quantiles=10, random_state=0, copy=True)
ifn_gene_data_quantile
```

Out[46]:
```
array([[0.59298164, 0.55166864, 0.63851285, ..., 0.85026896, 0.8005734 ,
        0.5555556 ],
       [0.8394851 , 0.91935915, 0.86708057, ..., 0.85026896, 0.7229731 ,
        0.6666667 ],
       [0.98264027, 0.92640454, 0.9502683 , ..., 0.6666667 , 0.84529036,
        0.93929327],
       ...,
       [0.8144286 , 0.8727078 , 0.8004218 , ..., 0.18550755, 0.6950128 ,
        0.90817696],
       [0.91511613, 0.8978357 , 0.8450689 , ..., 0.9138109 , 0.17889225,
        0.27803558],
       [0.7669621 , 0.9050758 , 0.7040663 , ..., 0.80020934, 0.91811866,
        0.8924204 ]], dtype=float32)
```

Plot PCA for above 25 gene expression data on top of histology of tumors

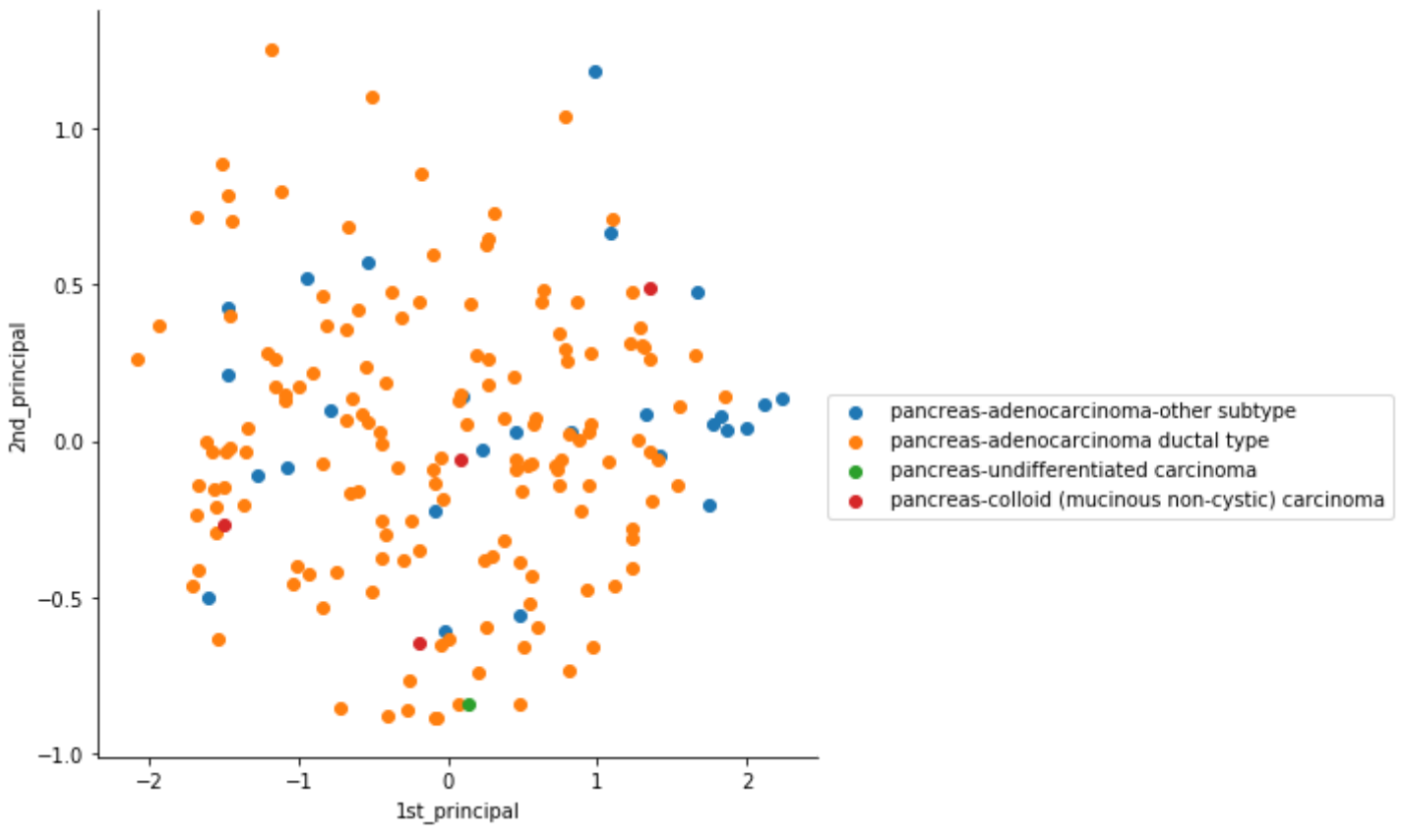
In [47]:
```
# initializing the pca
from sklearn.decomposition import PCA

# configuring the parameteres
# the number of components = 2
pcamodel = PCA(n_components=2)
pca = pcamodel.fit_transform(ifn_gene_data_quantile)

# pca_reduced will contain the 2-d projects of simple data
print("shape of pca_reduced.shape = ", pca.shape)
```

```
shape of pca_reduced.shape =  (183, 2)
```

For histological_type

In [48]:
```
# attaching the label for each 2-d data point
Y = column_meta_data_imputed.histological_type
pca_data = np.vstack((pca.T, Y)).T
print(pca_data.shape)
# creating a new data fram which help us in ploting the result data
pca_df = pd.DataFrame(data=pca_data, columns=("1st_principal", "2nd_principal", "label"))
sns.FacetGrid(pca_df, hue="label", height=6).map(plt.scatter, '1st_principal', '2nd_principal')
plt.legend(loc='best',bbox_to_anchor=(1, 0., 0.5, 0.5))
plt.show()
```

```
(183, 3)
```



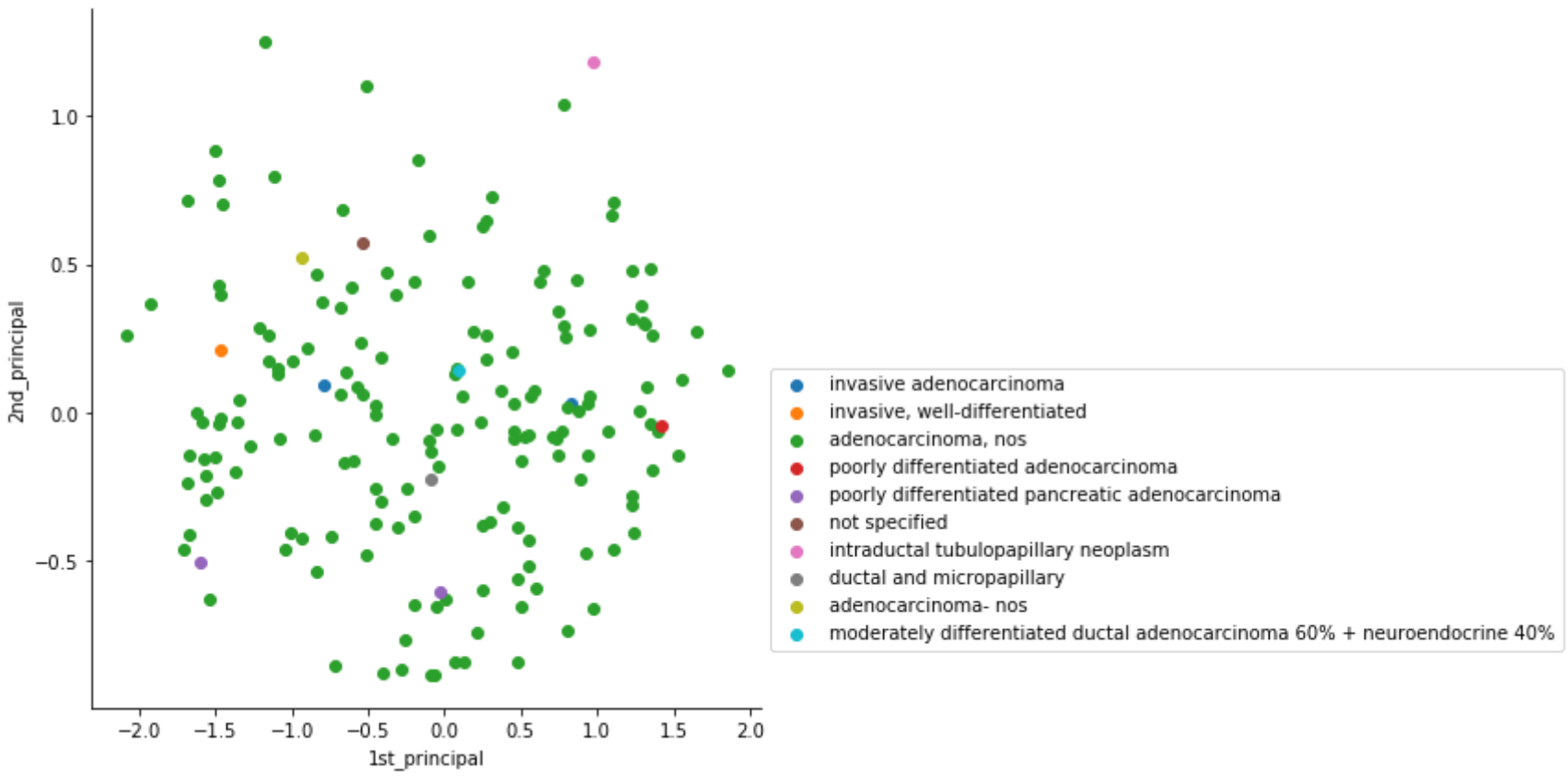
OBSERVATIONS:

- For these 25 genes of `Type 1 IFN signature` above plot shows most of `pancreatic-adenocarcinoma cancer is of ductal type` .

For `histological_type_other` : After removing Neuroendocrine tumors data

In [50]:
```
# attaching the label for each 2-d data point
Y = column_meta_data_neuro_removed.histological_type_other
pca_data = np.vstack((pca.T, Y)).T
print(pca_data.shape)
# creating a new data fram which help us in ploting the result data
pca_df = pd.DataFrame(data=pca_data, columns=("1st_principal", "2nd_principal", "label"))
sns.FacetGrid(pca_df, hue="label", height=6).map(plt.scatter, '1st_principal', '2nd_principal')
plt.legend(loc='best',bbox_to_anchor=(1, 0., 0.5, 0.5))
plt.show()
```

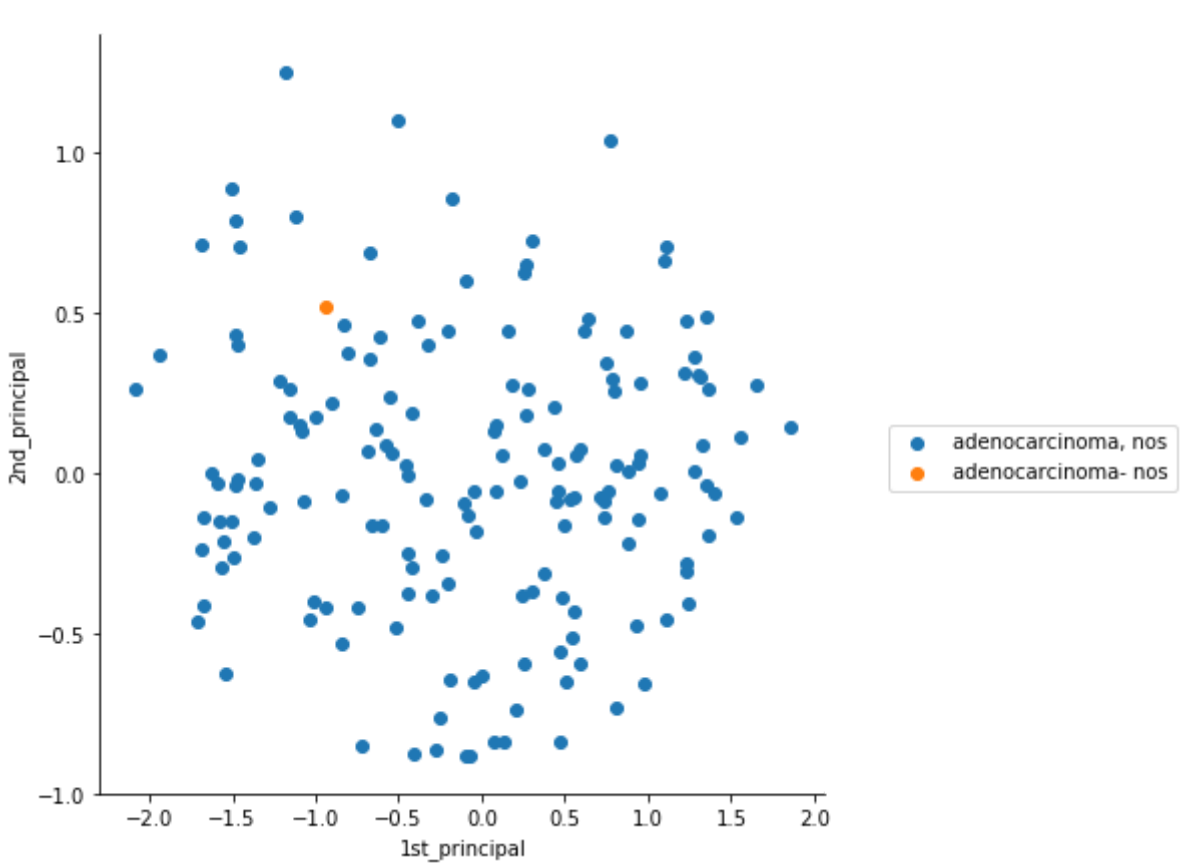
```
(183, 3)
```



For histological_type_other : After keep `only Adenocarcinoma NOS` tumors data

In [51]:
```
# attaching the label for each 2-d data point
Y = column_meta_data_adeno.histological_type_other
pca_data = np.vstack((pca.T, Y)).T
print(pca_data.shape)
# creating a new data fram which help us in ploting the result data
pca_df = pd.DataFrame(data=pca_data, columns=("1st_principal", "2nd_principal", "label"))
sns.FacetGrid(pca_df, hue="label", height=6).map(plt.scatter, '1st_principal', '2nd_principal')
plt.legend(loc='best',bbox_to_anchor=(1, 0., 0.5, 0.5))
plt.show()
```

```
(183, 3)
```



Applying `GSVA` [Gene Set variation Analysis]

In [52]:
```
#expression_df = row_meta_data_logScale.drop(columns = ifn_gene_list)
#expression_df = expression_df.T
expression_df = row_meta_data_logScale.T
print("Shape of expression data",expression_df.shape)

ifn_genes.columns = ["member"]
ifn_genes["name"] = ifn_genes.member
print("Shape of genes",ifn_genes.shape)
ifn_genes
```

```
Shape of expression data (18465, 183)
Shape of genes (25, 2)
```

Out[52]:

	member	name
0	IFIT1	IFIT1
1	IFI44	IFI44
2	IFIT3	IFIT3
3	MX2	MX2
4	OAS1	OAS1
5	OAS3	OAS3
6	BST2	BST2
7	IFITM1	IFITM1
8	MX1	MX1
9	STAT1	STAT1
10	IFI27	IFI27
11	CXCL10	CXCL10
12	IFI16	IFI16
13	IFI30	IFI30
14	IFIH1	IFIH1
15	IFIT2	IFIT2
16	IFITM2	IFITM2
17	IRF1	IRF1
18	IRF9	IRF9
19	IRGM	IRGM
20	ISG15	ISG15
21	OAS2	OAS2
22	PSME1	PSME1
23	SOCS1	SOCS1
24	STAT2	STAT2

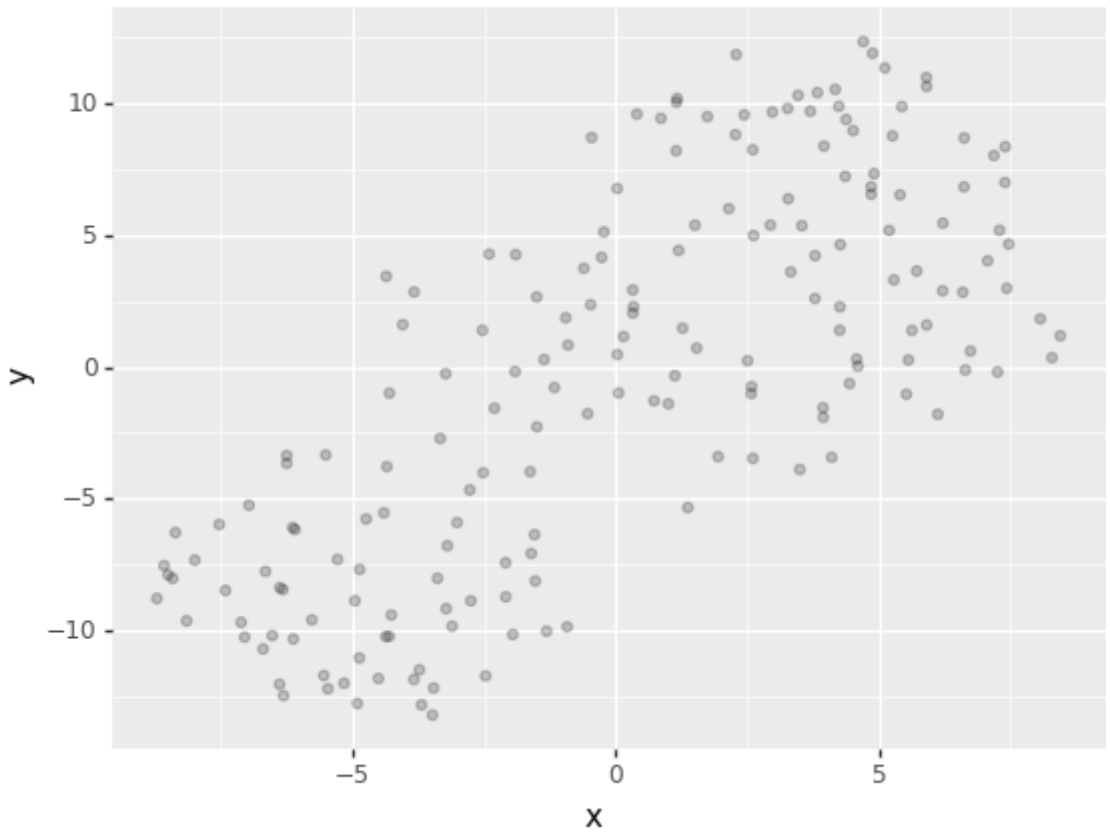
In [53]:
```
# Applying Gene Set variation Analysis (GSVA)
pathways_df = gsva(expression_df,ifn_genes)
pathways_df
```

Out[53]:

name	aab1-Primary solid Tumor	aab4-Primary solid Tumor	aab6-Primary solid Tumor	aab8-Primary solid Tumor	aab9-Primary solid Tumor	aaba-Primary solid Tumor	aabe-Primary solid Tumor	aabf-Primary solid Tumor	aabh-Primary solid Tumor	aabi-Primary solid Tumor	...	aauh-Primary solid Tumor	aaui-Primary solid Tumor	I
BST2	0.698549	0.921252	0.795602	0.448873	-0.867418	-0.480719	0.332539	0.169844	-0.910637	-0.305676	...	-0.405546	-0.560009	0
CXCL10	0.931001	0.805243	0.691183	0.147205	-0.215663	-0.957756	-0.755308	0.955373	-0.287912	-0.454073	...	-0.536937	-0.866984	-0
IFI16	-0.694541	-0.365360	0.800477	0.896447	-0.598029	-0.602903	0.926885	0.141681	-0.722812	-0.346295	...	0.167894	0.028921	-0
IFI27	0.766031	0.880741	0.895039	0.580914	-0.249242	-0.787695	0.205373	0.512782	-0.365035	-0.900997	...	-0.771231	-0.085789	0
IFI30	-0.295494	0.278163	0.159120	-0.041269	-0.189341	-0.786179	0.509099	0.870776	-0.078964	-0.483211	...	-0.612652	-0.587630	0
IFI44	0.168111	0.974437	0.914103	0.840013	-0.558601	-0.736244	-0.237543	0.983427	-0.565100	-0.655221	...	-0.794952	-0.911503	0
IFIH1	0.233752	0.909012	0.825607	0.818133	0.197682	-0.838388	0.356911	0.990035	0.067374	-0.341638	...	-0.683925	-0.670927	-0
IFIT1	0.319866	0.825065	0.973354	0.837955	-0.459055	-0.960789	-0.032604	0.983969	-0.371642	-0.706889	...	-0.567158	-0.943999	0

Visualize Whole Data after applying GSVA for 25 gene set using TSNE dimensonality reduction algorithm

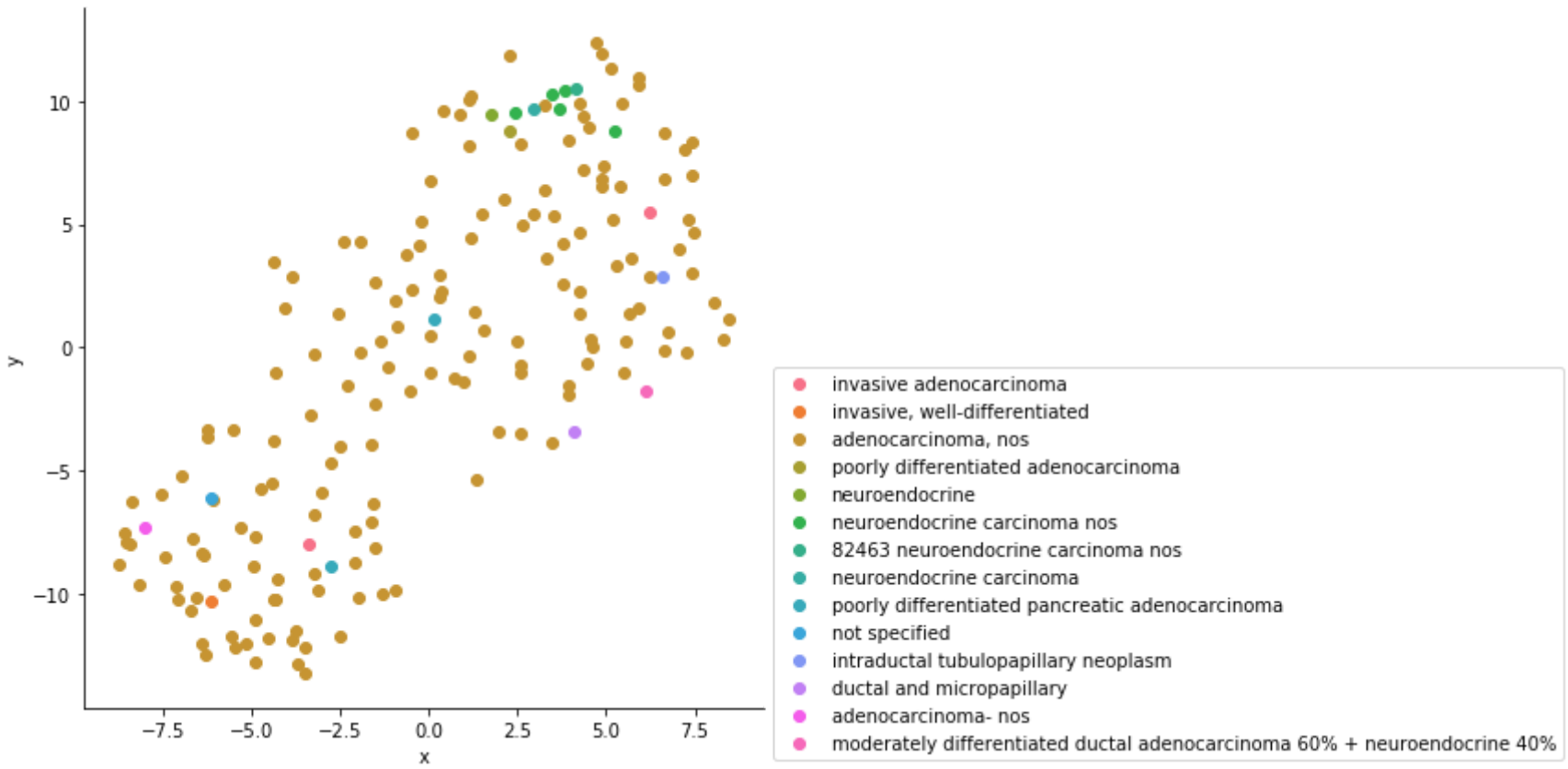
In [54]:
```
YV = TSNE(n_components=2).\
    fit_transform(pathways_df.T)
pf = pd.DataFrame(YV).rename(columns={0:'x',1:'y'})

(ggplot(pf,aes(x='x',y='y'))
 + geom_point(alpha=0.2)
)
```



Out[54]: <ggplot: (129877019221)>

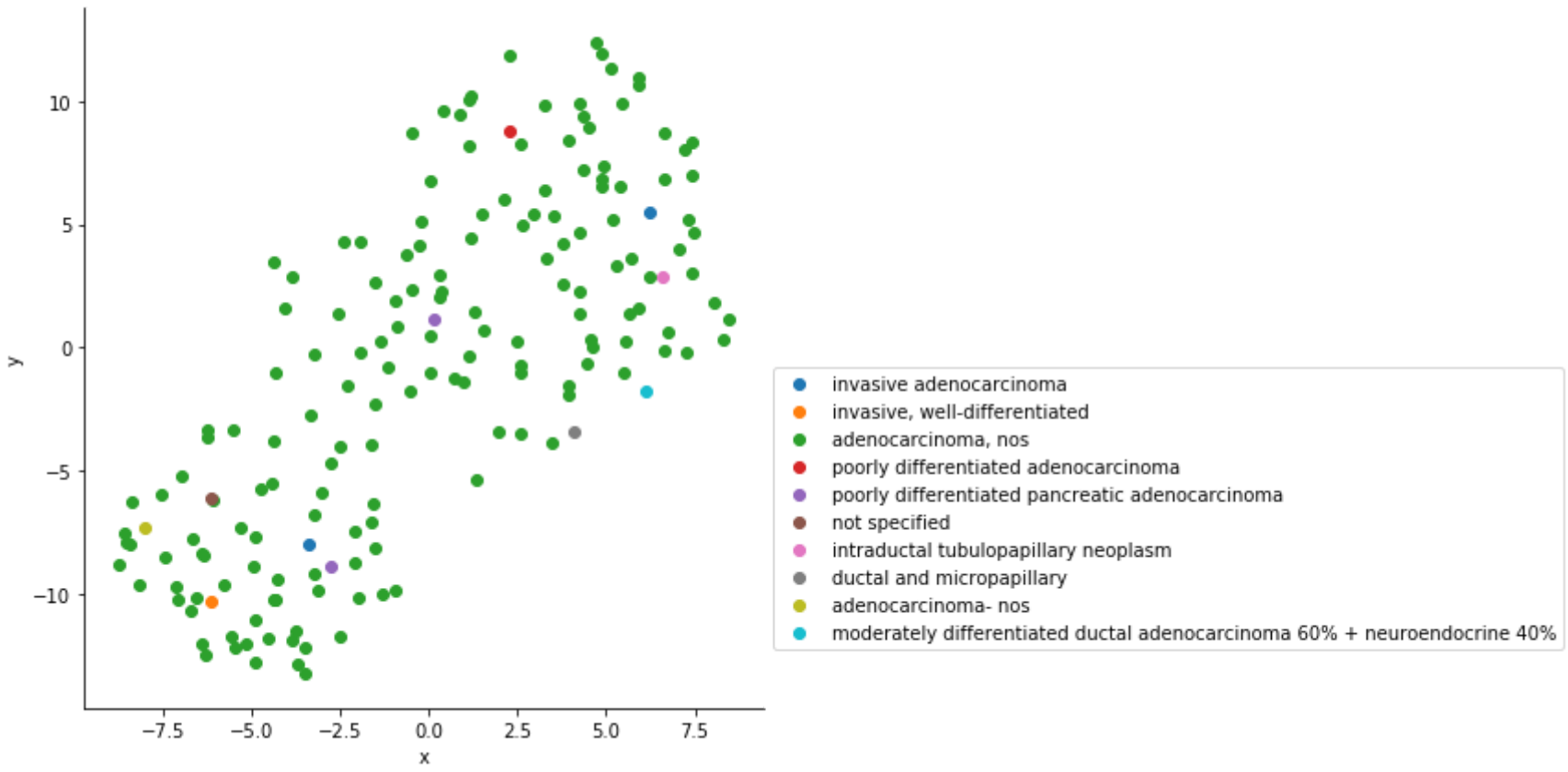
For histological_type_other including neuroendocrine tumors

In [55]:
```
Y = column_meta_data_imputed["histological_type_other"]
pca_data = np.vstack((YV.T, Y)).T
pf = pd.DataFrame(pca_data).rename(columns={0:'x',1:'y',2:'label'})
sns.FacetGrid(pf, hue="label", height=6).map(plt.scatter, 'x', 'y')
plt.legend(loc='best',bbox_to_anchor=(1, 0., 0.5, 0.5))
plt.show()
```



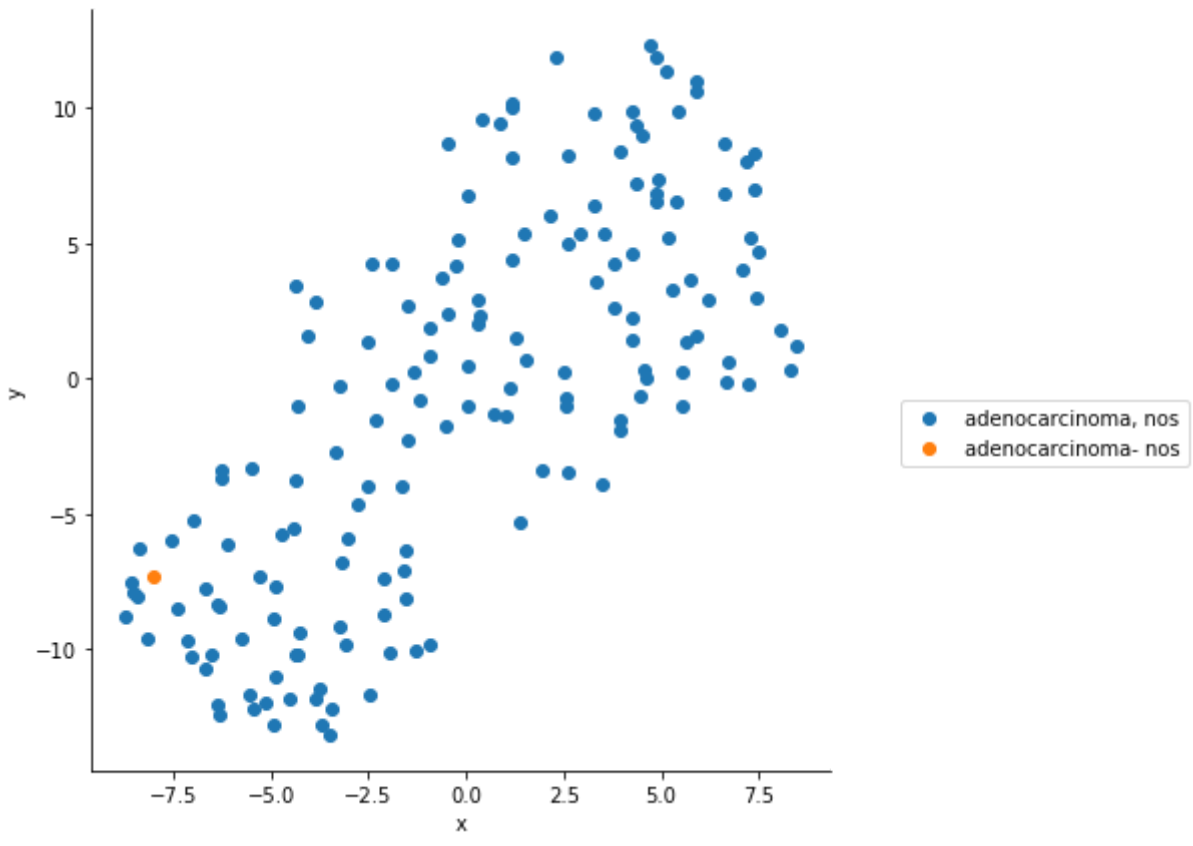
For histological_type_other excluding neuroendocrine tumors

In [56]:
```
Y = column_meta_data_neuro_removed["histological_type_other"]
pca_data = np.vstack((YV.T, Y)).T
pf = pd.DataFrame(pca_data).rename(columns={0:'x',1:'y',2:'label'})
sns.FacetGrid(pf, hue="label", height=6).map(plt.scatter, 'x', 'y')
plt.legend(loc='best',bbox_to_anchor=(1, 0., 0.5, 0.5))
plt.show()
```



For histological_type_other keep only adenocarcinoma nos tumors

In [57]:
```
Y = column_meta_data_adeno["histological_type_other"]
pca_data = np.vstack((YV.T, Y)).T
pf = pd.DataFrame(pca_data).rename(columns={0:'x',1:'y',2:'label'})
sns.FacetGrid(pf, hue="label", height=6).map(plt.scatter, 'x', 'y')
plt.legend(loc='best',bbox_to_anchor=(1, 0., 0.5, 0.5))
plt.show()
```



OBSERVATIONS:

- Most of type of tumors are adenocarcinoma-nos type

Null values in column data(adeno only) in histological_type_other.

In [71]:
```
1 column_meta_data_adeno.histological_type_other.isnull().sum()
```

Out[71]: 18

Negative values in `IFN` genes

In [73]:
```
1 data = meta_data.data_df.T[~column_meta_data_adeno.histological_type_other.isnull()][ifn_gene_list]
```

In [74]:
```
1 (data[ifn_gene_list].isnull().sum() + (data[ifn_gene_list]<= 0).sum())
```

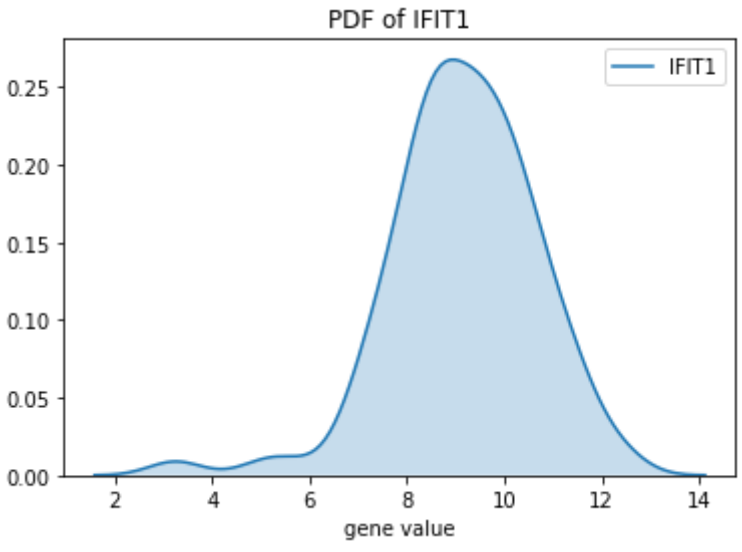
Out[74]:
```
rid
IFIT1       0
IFI44       0
IFIT3       0
MX2         0
OAS1        0
OAS3        0
BST2        0
IFITM1      0
MX1         0
STAT1       0
IFI27       0
CXCL10      1
IFI16       0
IFI30       0
IFIH1       0
IFIT2       0
IFITM2      0
IRF1        0
IRF9        0
IRGM      107
ISG15       0
OAS2        0
PSME1       0
SOCS1       0
STAT2       0
dtype: int64
```

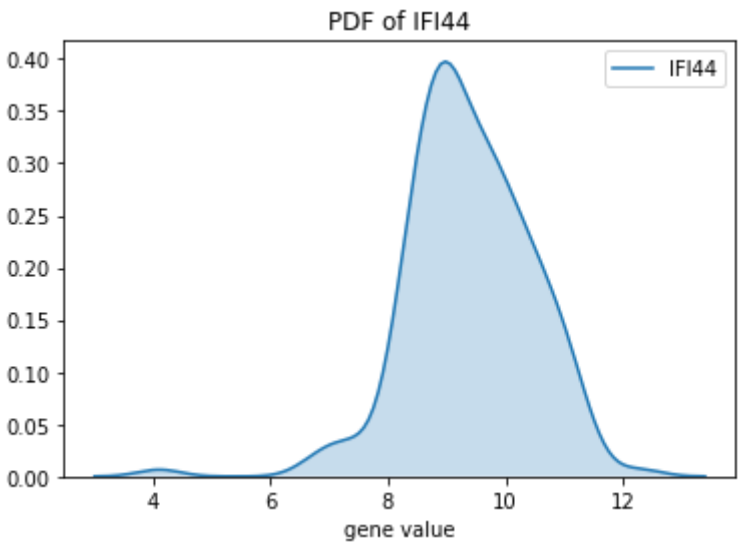
OBSERVATIONS:

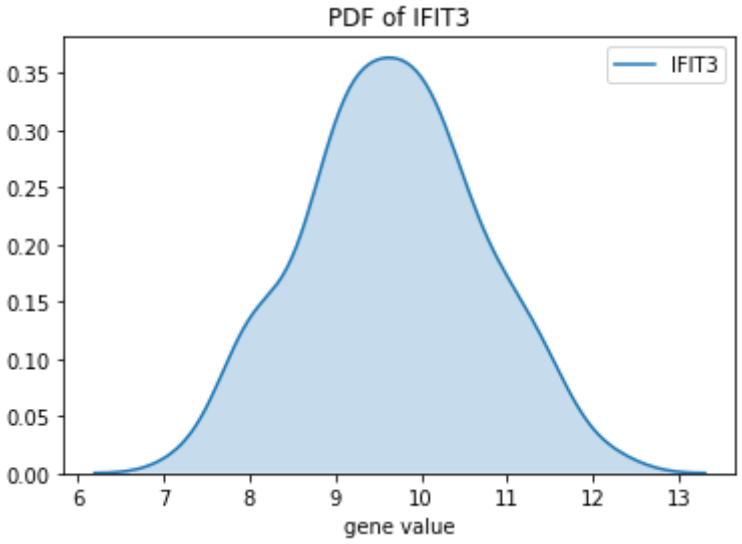
- `IRGM` gene of IFN type have `nan` and `negative values` .
- `CXCL10` has 2 negative values

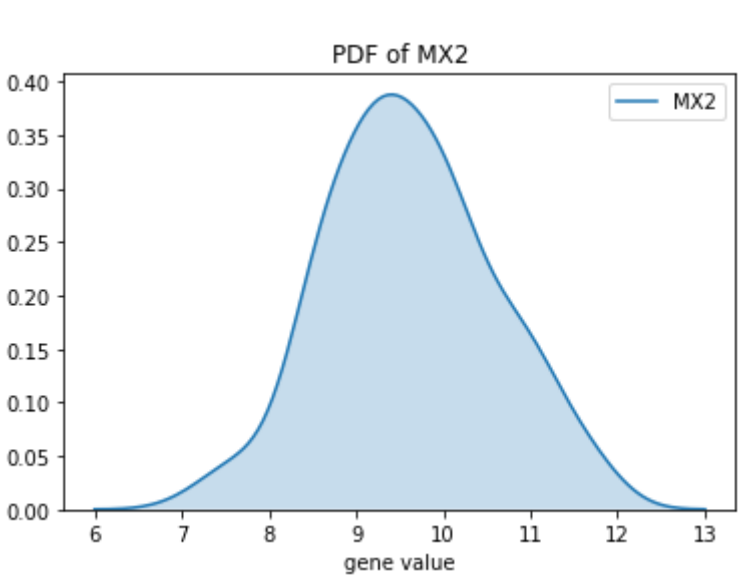
`PDF of all gene` present in `IFN`

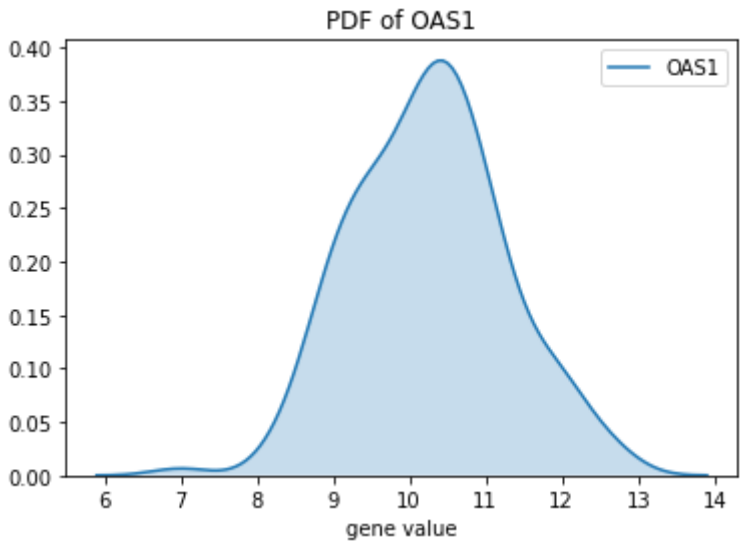
In [75]:
```
for col in data.columns:
    if col in ifn_gene_list:
        sns.kdeplot(data[col],shade=True)
        plt.xlabel('gene value')
        plt.title('PDF of '+col);
        plt.show();
```

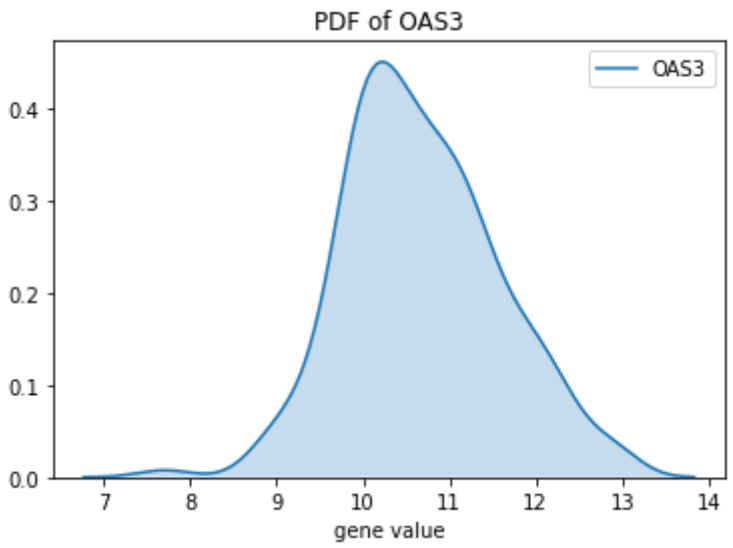


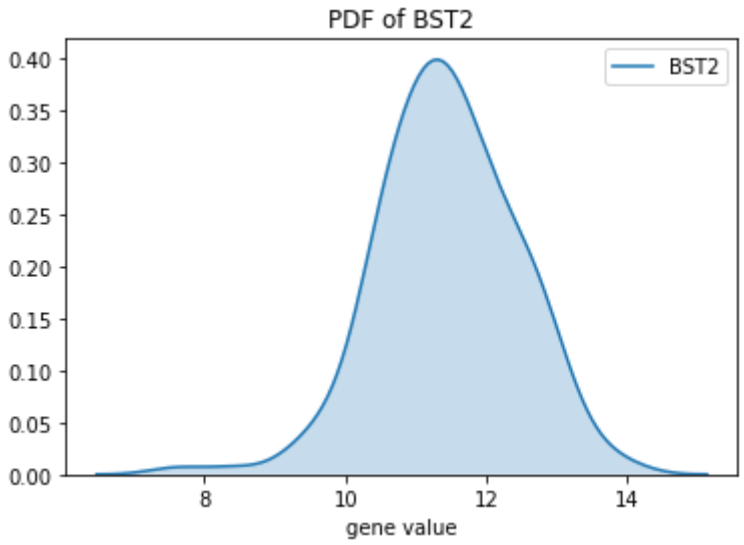


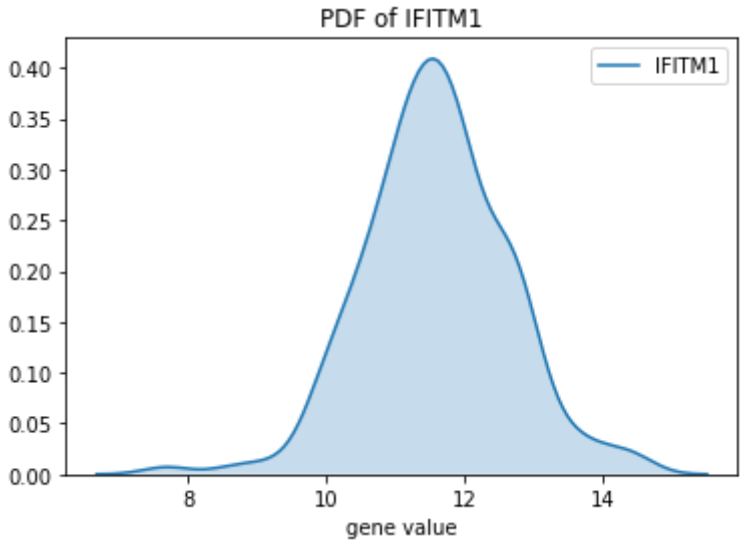


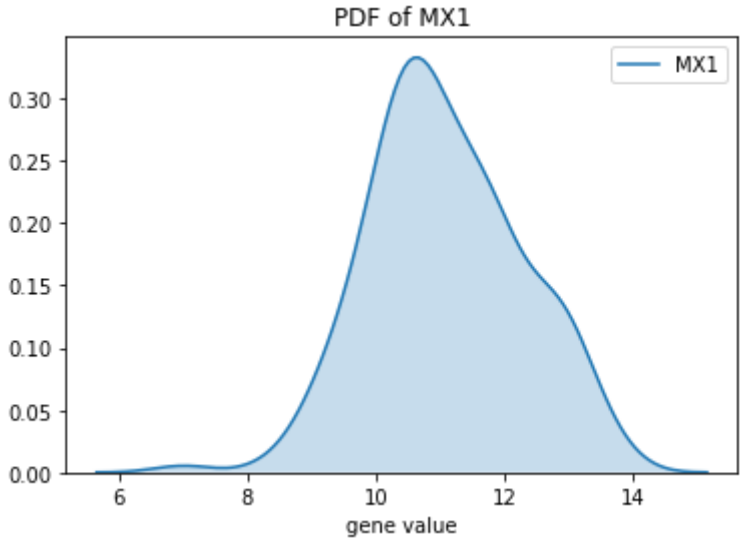


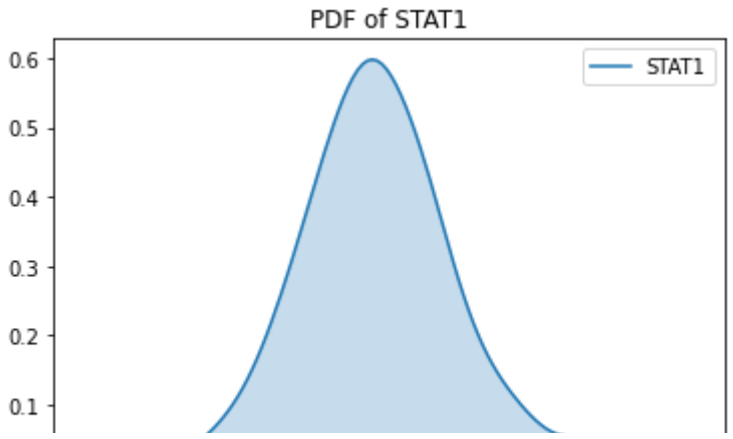

PDF of OAS1
OAS1
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
6
7
8
9
10
11
12
13
14
gene value


PDF of OAS3
OAS3
0.4
0.3
0.2
0.1
0.0
7
8
9
10
11
12
13
14
gene value

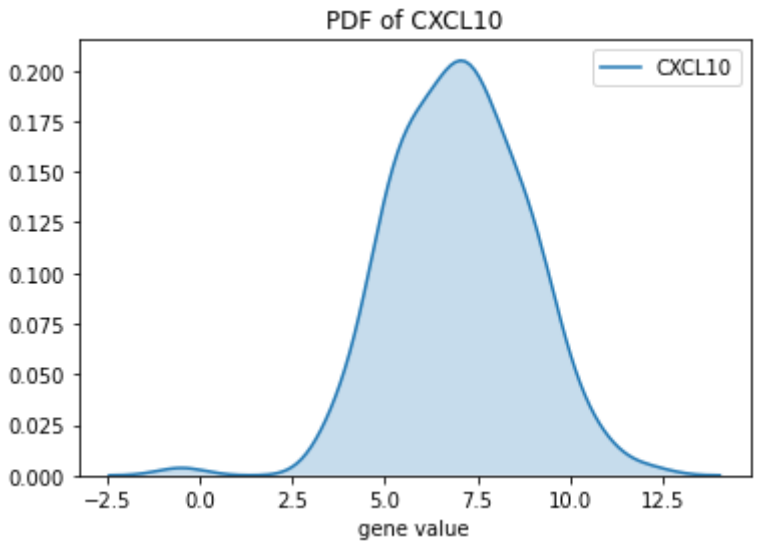

PDF of BST2
BST2
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
8
10
12
14
gene value

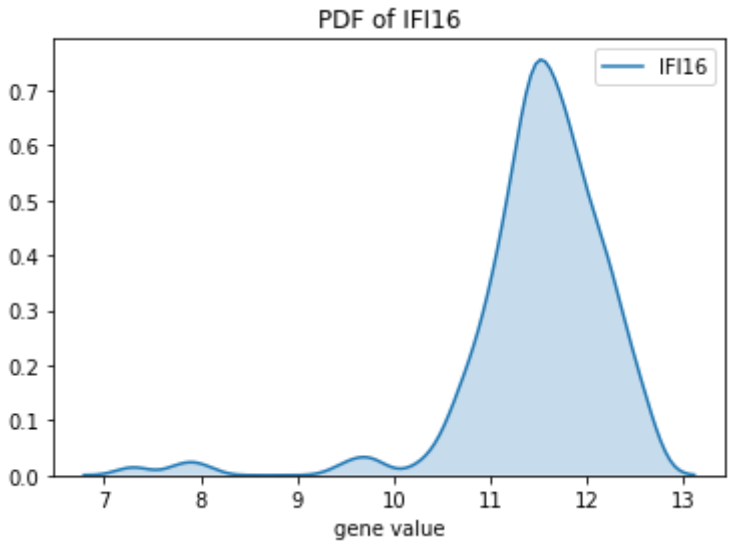

PDF of IFITM1
IFITM1
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
8
10
12
14
gene value

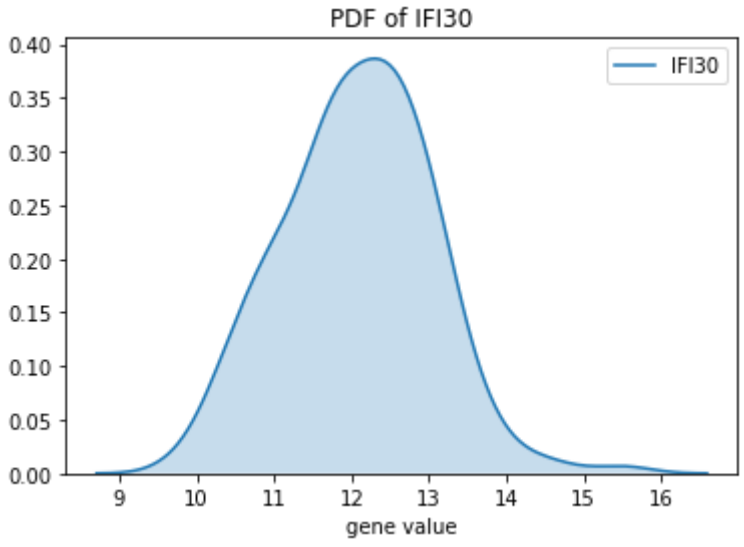

PDF of MX1
MX1
0.30
0.25
0.20
0.15
0.10
0.05
0.00
6
8
10
12
14
gene value

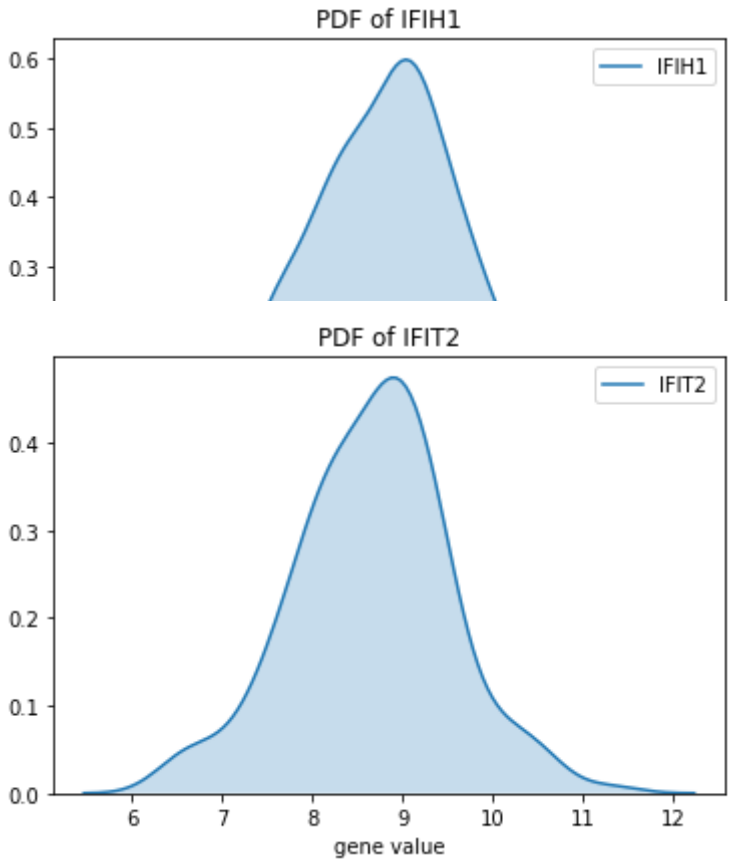

PDF of STAT1
STAT1
0.6
0.5
0.4
0.3
0.2
0.1

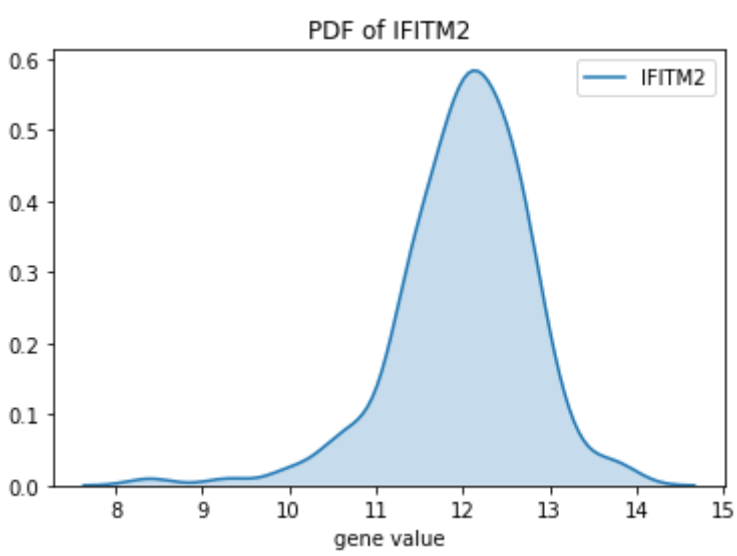
PDF of IFI27
IFI27
0.25
0.20
0.15
0.10
0.05
0.00
8
10
12
14
16
gene value

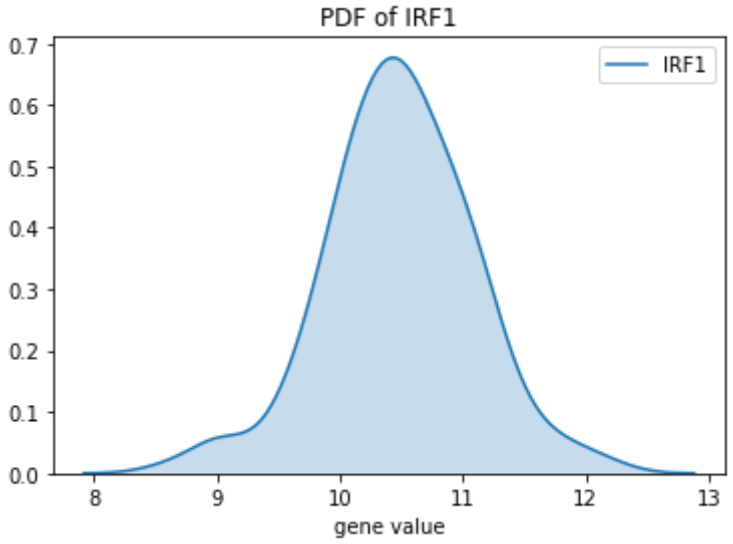

PDF of CXCL10
CXCL10
0.200
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
−2.5
0.0
2.5
5.0
7.5
10.0
12.5
gene value

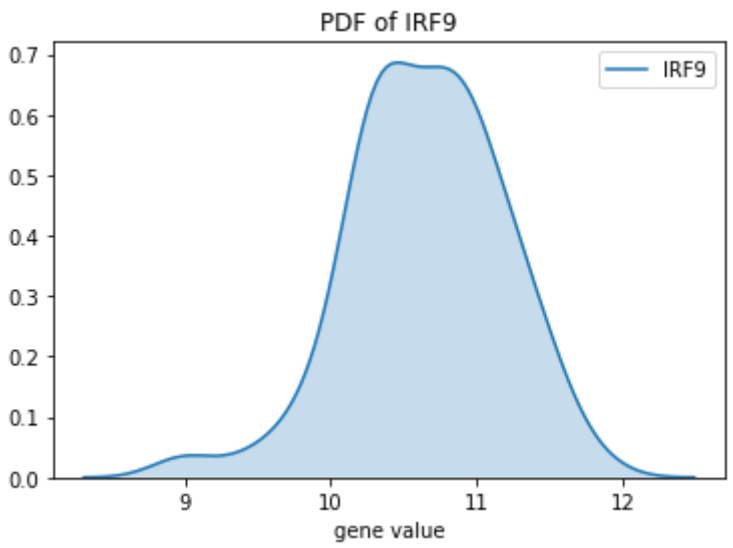

PDF of IFI16
IFI16
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
7
8
9
10
11
12
13
gene value


PDF of IFI30
IFI30
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
9
10
11
12
13
14
15
16
gene value

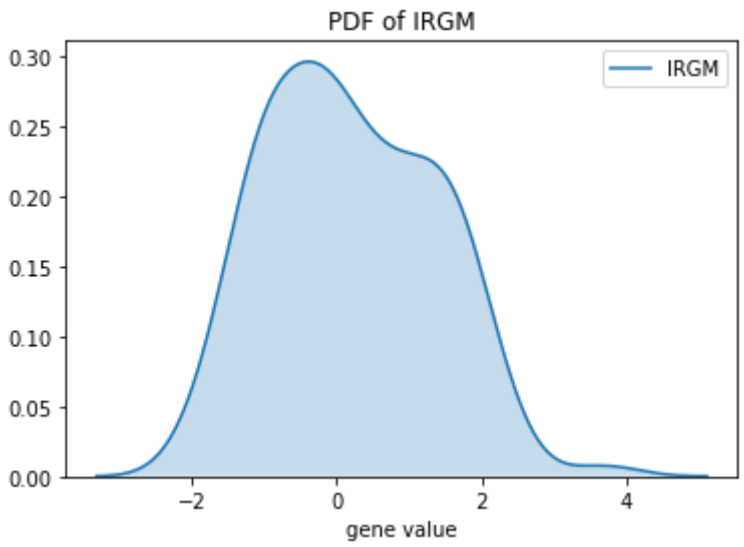


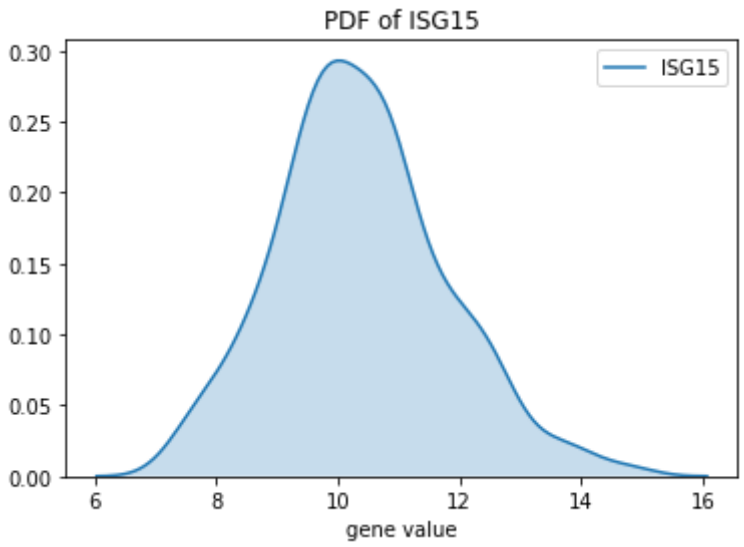


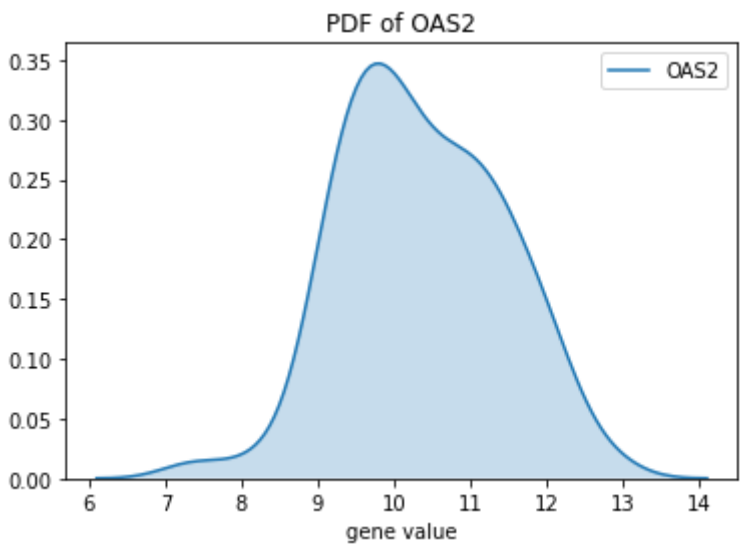


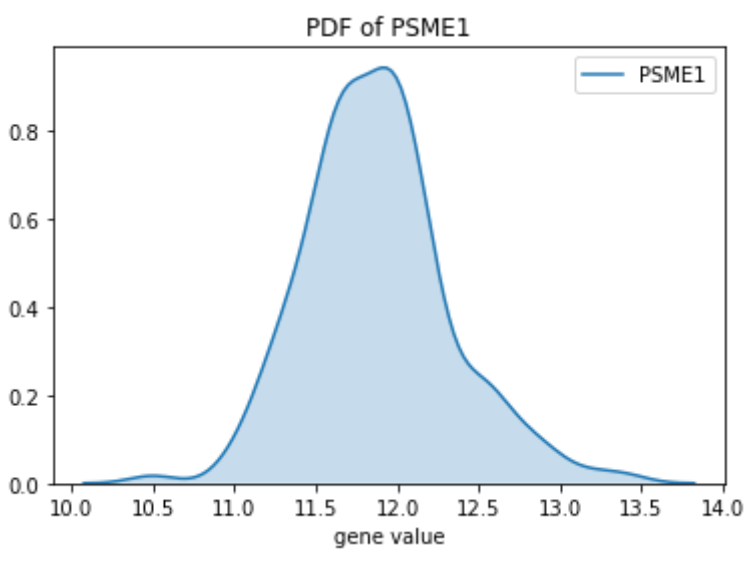


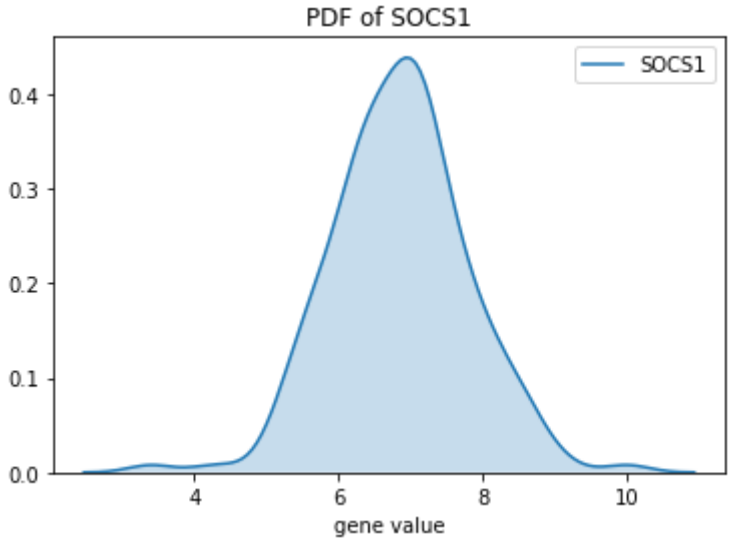

```
C:\Users\Tarun Makkar\AppData\Roaming\Python\Python37\site-packages\statsmodels\nonparametric\kde.py:547: RuntimeWarnin
g: invalid value encountered in greater
C:\Users\Tarun Makkar\AppData\Roaming\Python\Python37\site-packages\statsmodels\nonparametric\kde.py:547: RuntimeWarnin
g: invalid value encountered in less
```

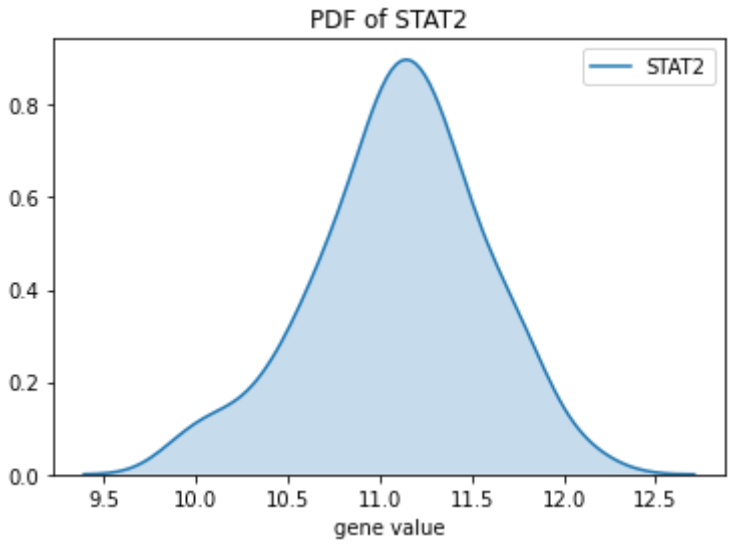

PDF of IRGM
IRGM
0.30
0.25
0.20
0.15
0.10
0.05
0.00
−2
0
2
4
gene value


PDF of ISG15
ISG15
0.30
0.25
0.20
0.15
0.10
0.05
0.00
6
8
10
12
14
16
gene value


PDF of OAS2
OAS2
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
6
7
8
9
10
11
12
13
14
gene value


PDF of PSME1
PSME1
0.8
0.6
0.4
0.2
0.0
10.0
10.5
11.0
11.5
12.0
12.5
13.0
13.5
14.0
gene value


PDF of SOCS1
SOCS1
0.4
0.3
0.2
0.1
0.0
4
6
8
10
gene value



OBSERVATIONS:

- All genes show similar observations except `IRGM` and `CXCL10`