

# Steady-State Ligand-Receptor Inference

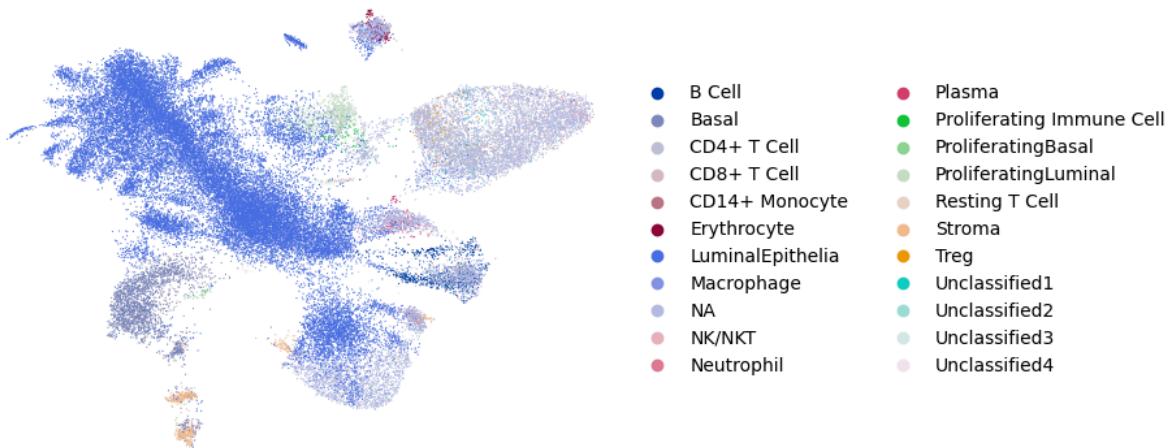
## LOADING DATA

```
In [2]: # import packages
import liana as li
import scanpy as sc
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
import os
```

```
In [3]: # load RNAseq data
data = sc.read_h5ad("/Users/eunicelee153/Desktop/WORK/CLINICAL/Angelo Lab/DC
```

```
In [4]: # data umap
sc.pl.umap(data, color='subcluster', title='', frameon=False)
```

```
... storing 'orig.ident' as categorical
... storing 'Phase' as categorical
... storing 'old.ident' as categorical
... storing 'DoubletFinder' as categorical
... storing 'ARTCLASS' as categorical
... storing 'subcluster' as categorical
```



```
In [5]: # normalize counts
data.raw.X
```

```
Out[5]: <45594x38224 sparse matrix of type '<class 'numpy.float64'>'>
with 90753307 stored elements in Compressed Sparse Row format>
```

# METHODS

```
In [6]: # import all individual methods
from liana.method import singlecellsignalr, connectome, cellphonedb, natmi,
```

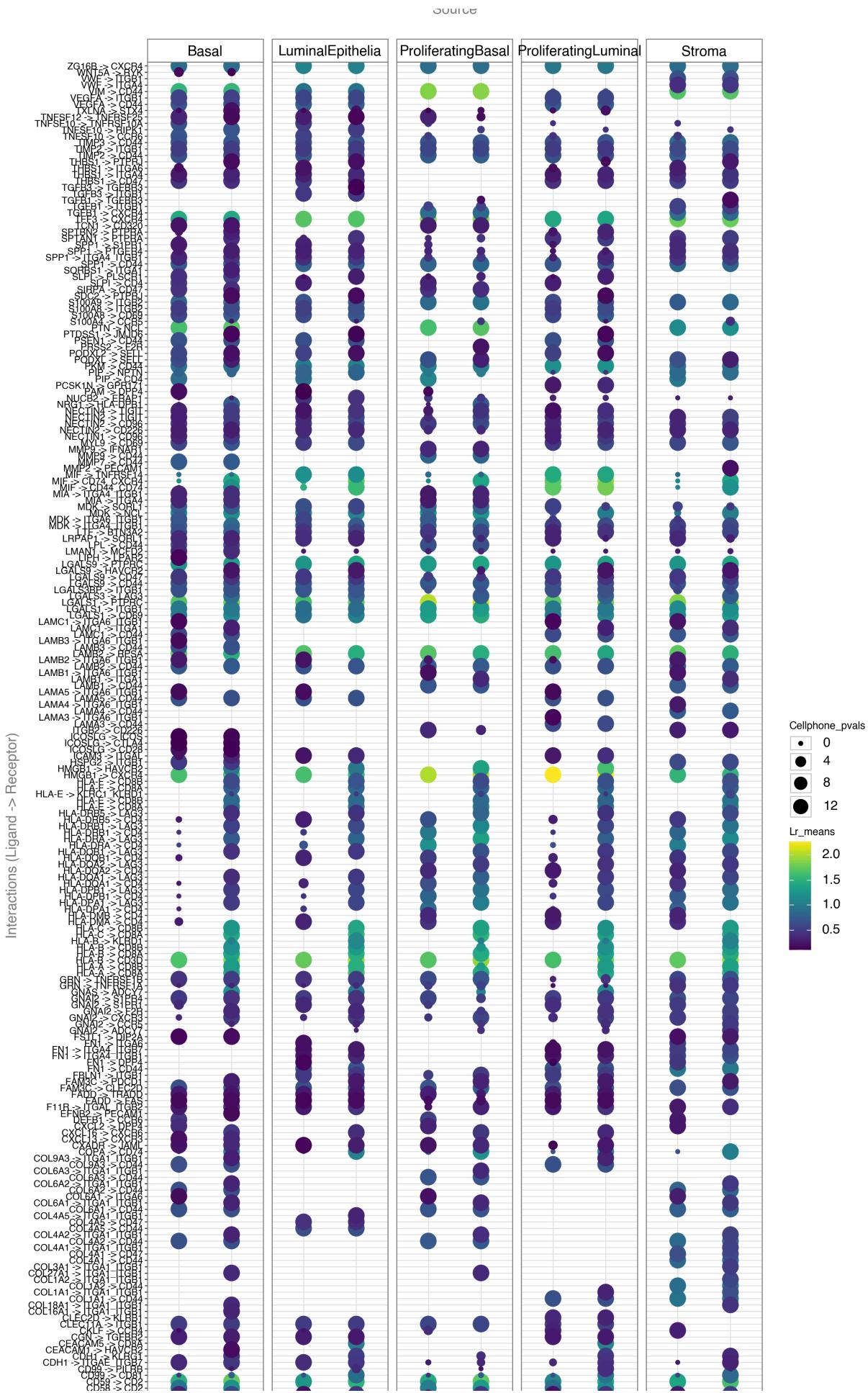
## Method: CellphoneDB

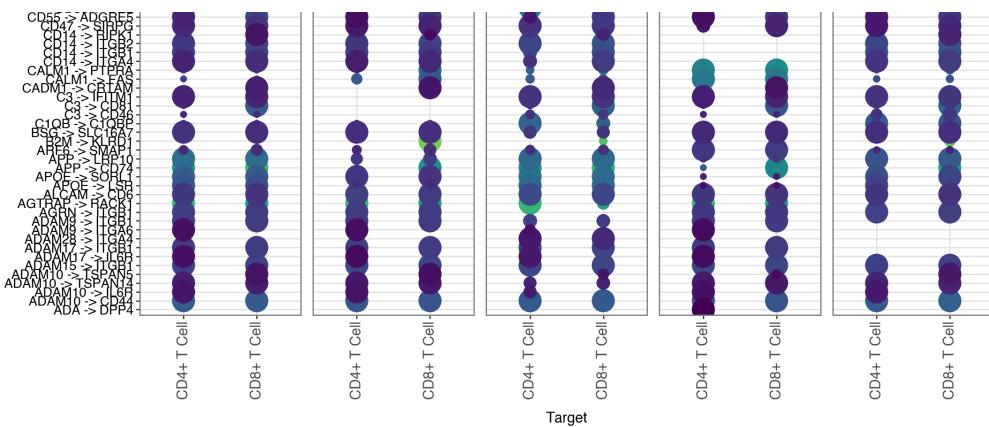
```
In [7]: # run cellphonedb
cellphonedb(data,
             groupby='subcluster',
             resource_name='consensus',
             expr_prop=0.1,
             verbose=True, key_added='cpdb_res')
```

```
Using resource `consensus`.
Using `raw`!
/Users/eunicelee153/fsl/lib/python3.11/site-packages/anndata/_core/anndata.py:401: FutureWarning: The dtype argument is deprecated and will be removed in late 2024.
5442 features of mat are empty, they will be removed.
/Users/eunicelee153/fsl/lib/python3.11/site-packages/liana/method/_pipe_utils/_pre.py:150: ImplicitModificationWarning: Trying to modify attribute `.obs` of view, initializing view as actual.
/Users/eunicelee153/fsl/lib/python3.11/site-packages/liana/method/_pipe_utils/_pre.py:153: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.
0.04 of entities in the resource are missing from the data.
Generating ligand-receptor stats for 45594 samples and 1793 features
100%|██████████| 1000/1000 [00:36<00:00, 27.20it/s]
```

```
In [ ]: data.uns['cpdb_res'].head()
```

```
In [19]: # dotplot
li.pl.dotplot(adata = data,
               colour='lr_means',
               size='cellphone_pvals',
               inverse_size=True, # we inverse sign since we want small p-val
               source_labels=['LuminalEpithelia', 'Basal', 'Stroma', 'Proliferative'],
               target_labels=['CD4+ T Cell', 'CD8+ T Cell'],
               figure_size=(13, 25),
               # since cpdbv2 suggests using a filter to FPs, we filter the p-values
               filter_fun=lambda x: x['cellphone_pvals'] <= 0.05,
               uns_key='cpdb_res'
)
```

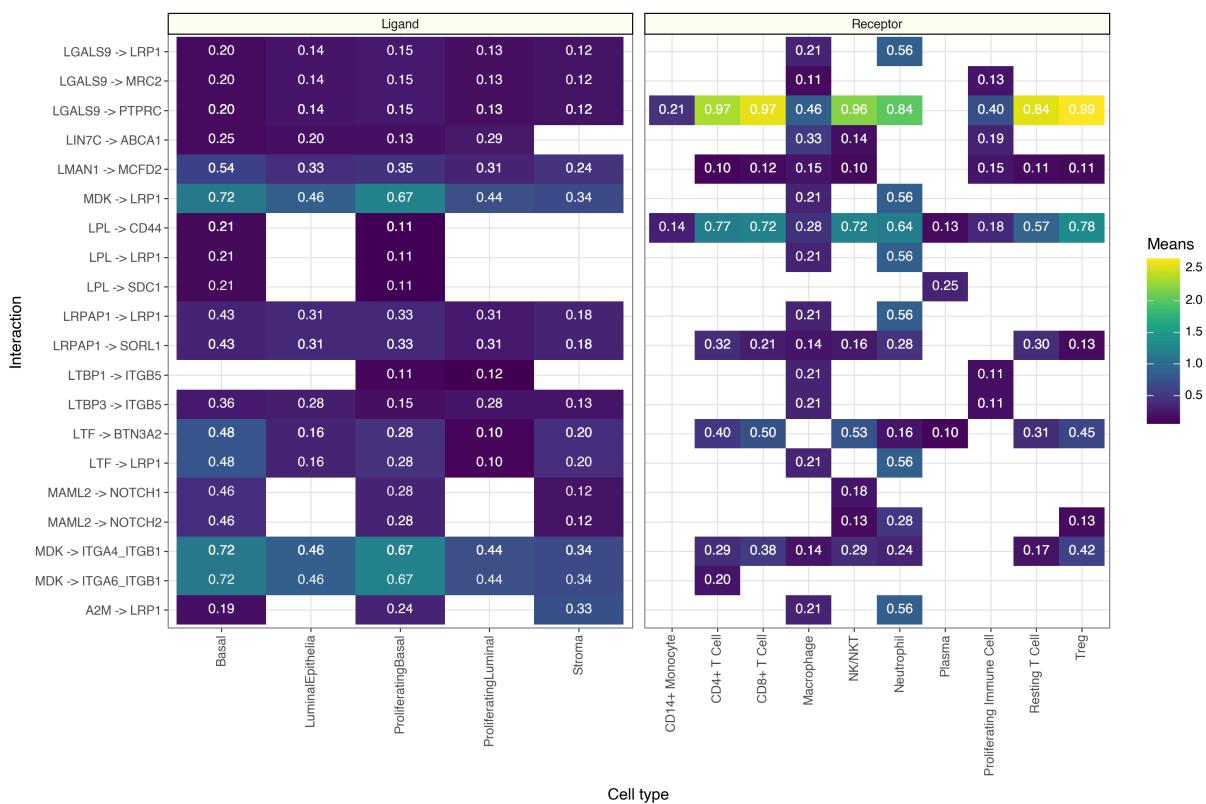




```
In [20]: my_plot = li.pl.tileplot(adata = data,
                               fill='means',
                               label='props',
                               label_fun=lambda x: f'{x:.2f}',
                               top_n=20,
                               orderby='cellphone_pvals',
                               orderby_ascending=True,
                               source_labels=['LuminalEpithelia', 'Basal', 'Stroma'],
                               target_labels=['CD4+ T Cell', 'CD8+ T Cell', 'CD14+ T Cell'],
                               uns_key='cpdb_res',
                               source_title='Ligand',
                               target_title='Receptor',
                               figure_size=(15, 10)
                               )
my_plot
```

/Users/eunicelee153/fsl/lib/python3.11/site-packages/liana/plotting/\_common.py:104: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame.  
Try using .loc[row\_indexer,col\_indexer] = value instead

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)



## Method: CellChat

```
In [22]: # run celchat
cellchat(adata = data,
         groupby='subcluster',
         resource_name='consensus',
         expr_prop=0.1,
         verbose=True, key_added='chat_res')
```

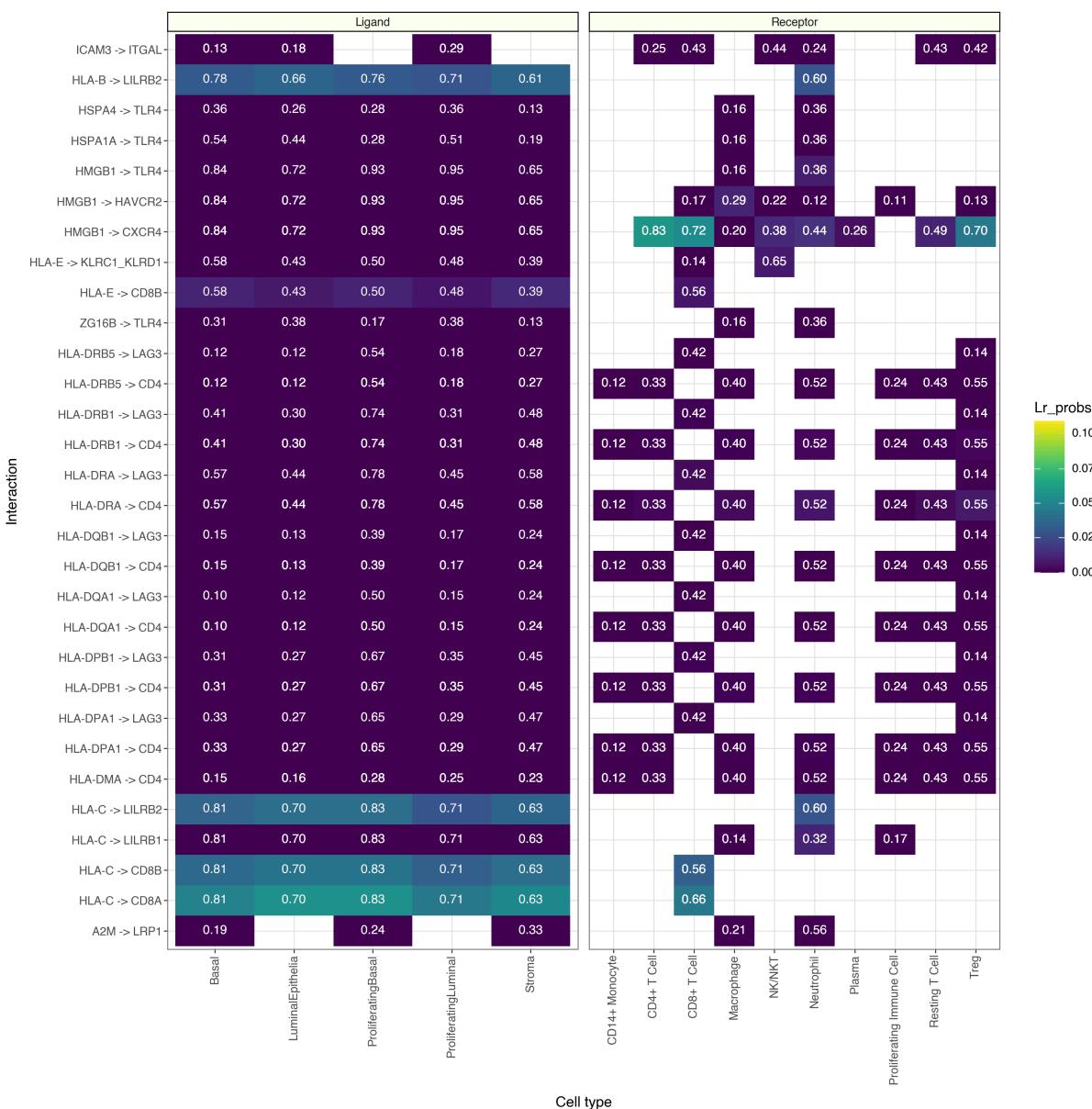
```
Using resource `consensus`.
Using `.raw`!
/Users/eunicelee153/fsl/lib/python3.11/site-packages/anndata/_core/anndata.py:401: FutureWarning: The dtype argument is deprecated and will be removed in late 2024.
5442 features of mat are empty, they will be removed.
/Users/eunicelee153/fsl/lib/python3.11/site-packages/liana/method/_pipe_utils/_pre.py:150: ImplicitModificationWarning: Trying to modify attribute `.obs` of view, initializing view as actual.
/Users/eunicelee153/fsl/lib/python3.11/site-packages/liana/method/_pipe_utils/_pre.py:153: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.
0.04 of entities in the resource are missing from the data.
Generating ligand-receptor stats for 45594 samples and 1793 features
100%|██████████| 1000/1000 [27:51<00:00, 1.67s/it]
```

```
In [ ]: # tileplot (tumor->immune)
cellchat_tileplot_1 = li.pl.tileplot(
```

```
adata=data,
fill='lr_probs', # Use interaction probabilities as fill values
label='props', # Use ligand proportions as labels
label_fun=lambda x: f'{x:.2f}',
top_n=30,
orderby='cellchat_pvals', # Order interactions by p-value
orderby_ascending=True, # Show the most significant interactions first
source_labels=['LuminalEpithelia', 'Basal', 'Stroma', 'ProliferatingBasal',
target_labels=['CD4+ T Cell', 'CD8+ T Cell', 'CD14+ Monocyte', 'NK/NKT',
uns_key='chat_res',
source_title='Ligand',
target_title='Receptor',
figure_size=(15, 15)
)
cellchat_tileplot_1
```

/Users/eunicelee153/fsl/lib/python3.11/site-packages/liana/plotting/\_common.py:104: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame.  
Try using .loc[row\_indexer,col\_indexer] = value instead

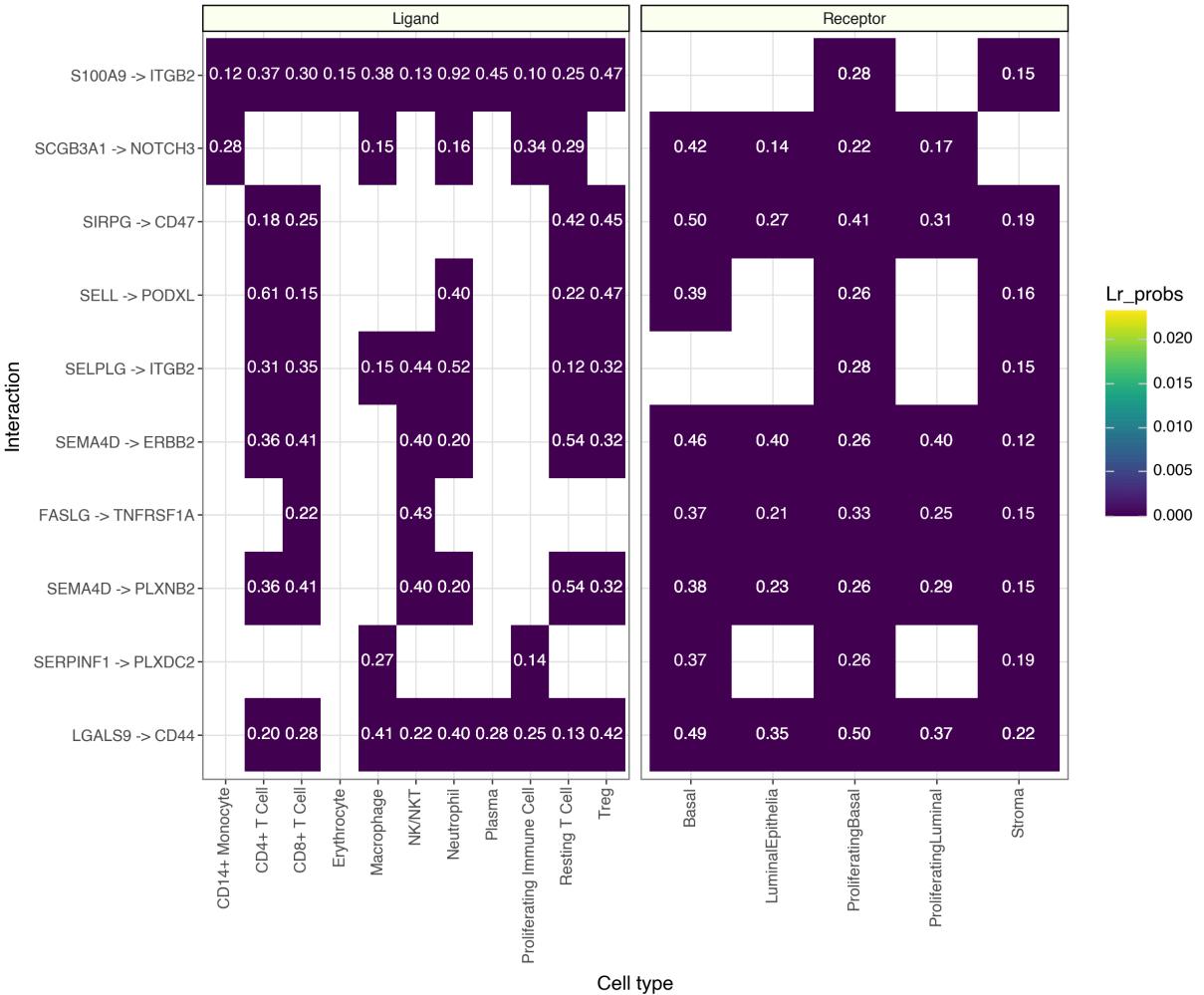
See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)



```
In [24]: # tileplot (immune->tumor)
cellchat_tileplot_2 = li.pl.tileplot(
    adata=data,
    fill='lr_probs', # Use interaction probabilities as fill values
    label='props', # Use ligand proportions as labels
    label_fun=lambda x: f'{x:.2f}',
    top_n=10,
    orderby='cellchat_pvals', # Order interactions by p-value
    orderbyAscending=True, # Show the most significant interactions first
    source_labels=['CD4+ T Cell', 'CD8+ T Cell', 'CD14+ Monocyte', 'NK/NKT',
    target_labels=['LuminalEpithelia', 'Basal', 'Stroma', 'ProliferatingBasal',
    uns_key='chat_res',
    source_title='Ligand',
    target_title='Receptor',
    figure_size=(12, 10)
)
cellchat_tileplot_2
```

```
/Users/eunicelee153/fsl/lib/python3.11/site-packages/liana/plotting/_common.py:104: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```



```
In [27]: # clean data
tumor_cells = ['LuminalEpithelia', 'Basal', 'Stroma', 'ProliferatingBasal', 'ProliferatingLuminal']
immune_cells = ['CD4+ T Cell', 'CD8+ T Cell', 'CD14+ Monocyte', 'NK/NKT', 'Neutrophil', 'Plasma', 'Proliferating Immune Cell', 'Resting T Cell', 'Treg']

filtered_data = data.ens['chat_res'][
    (data.ens['chat_res']['source'].isin(tumor_cells)) &
    (data.ens['chat_res']['target'].isin(immune_cells))
]

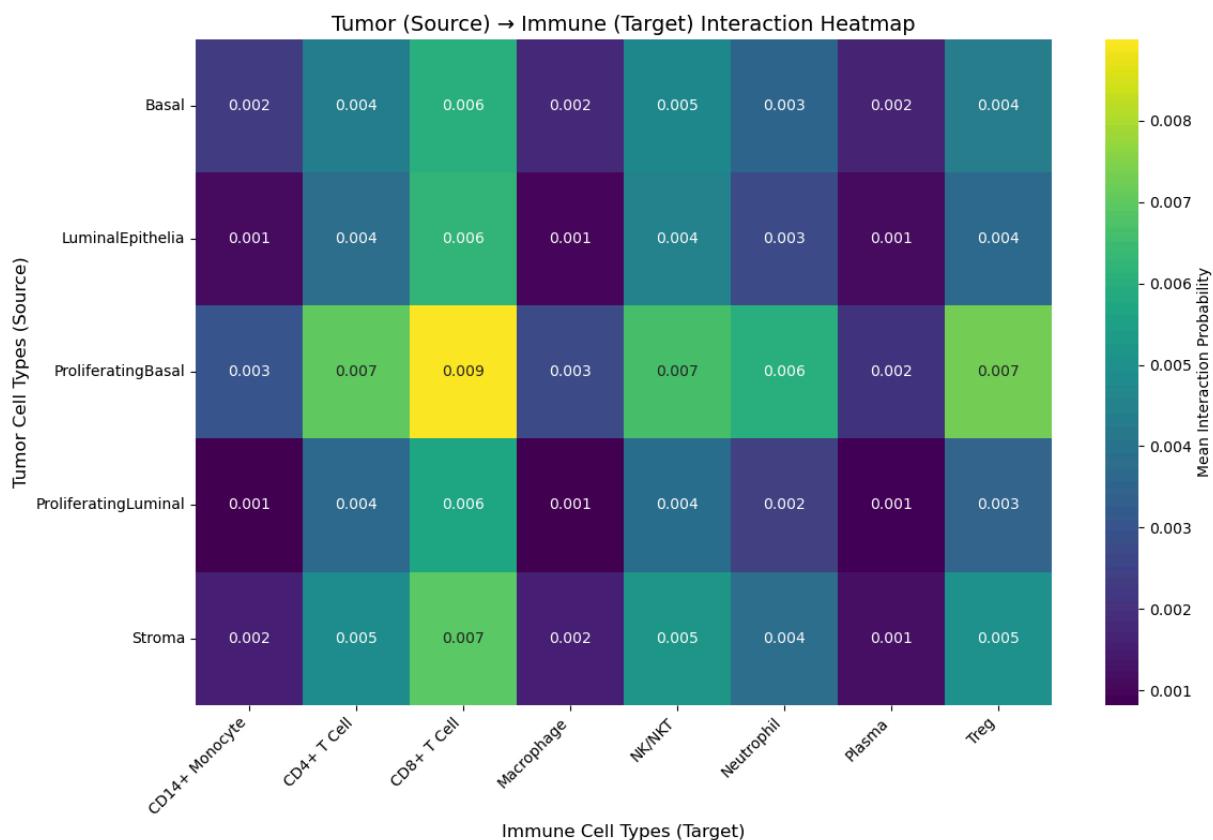
# heat map
heatmap_data = filtered_data.pivot_table(
    index='source', columns='target', values='lr_probs', aggfunc='mean'
)

plt.figure(figsize=(12, 8))
sns.heatmap(heatmap_data, cmap='viridis', annot=True, fmt=".3f", cbar_kws={'label': 'Lr_probs'})
plt.title("Tumor (Source) → Immune (Target) Interaction Heatmap", fontsize=14)
plt.xlabel("Immune Cell Types (Target)", fontsize=12)
```

```

plt.ylabel("Tumor Cell Types (Source)", fontsize=12)
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()

```



```

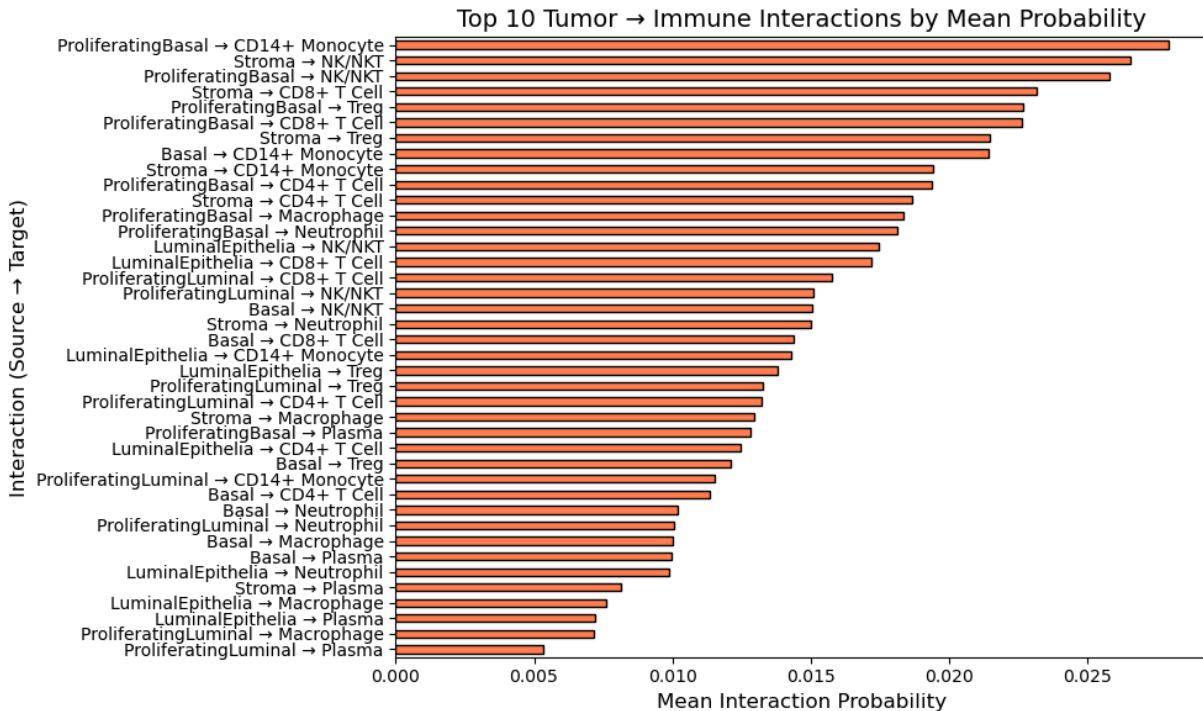
In [ ]: # clean data
edges = data.uns['chat_res'][[['source', 'target', 'lr_probs']]]
edges = edges.dropna()
edges['lr_probs'] = pd.to_numeric(edges['lr_probs'], errors='coerce') # Ens
edges = edges[edges['lr_probs'] > 0] # Keep only positive interaction proba
edges['source'] = edges['source'].astype(str)
edges['target'] = edges['target'].astype(str)

# tumor-immune cells; interaction pairs
tumor_cells = ['LuminalEpithelia', 'Basal', 'Stroma', 'ProliferatingBasal',
immune_cells = ['CD4+ T Cell', 'CD8+ T Cell', 'CD14+ Monocyte', 'NK/NKT', 'N
edges = edges[edges['source'].isin(tumor_cells) & edges['target'].isin(immu
edges['interaction'] = edges['source'] + " → " + edges['target']
interaction_scores = edges.groupby('interaction')['lr_probs'].mean().sort_val
top_interactions = interaction_scores.head(50)

# sideways barplot (tumor→immune)
plt.figure(figsize=(10, 6))
top_interactions.sort_values().plot(
    kind='barh', color='coral', edgecolor='black'
)
plt.title("Top 10 Tumor → Immune Interactions by Mean Probability", fontsize
plt.xlabel("Mean Interaction Probability", fontsize=12)
plt.ylabel("Interaction (Source → Target)", fontsize=12)

```

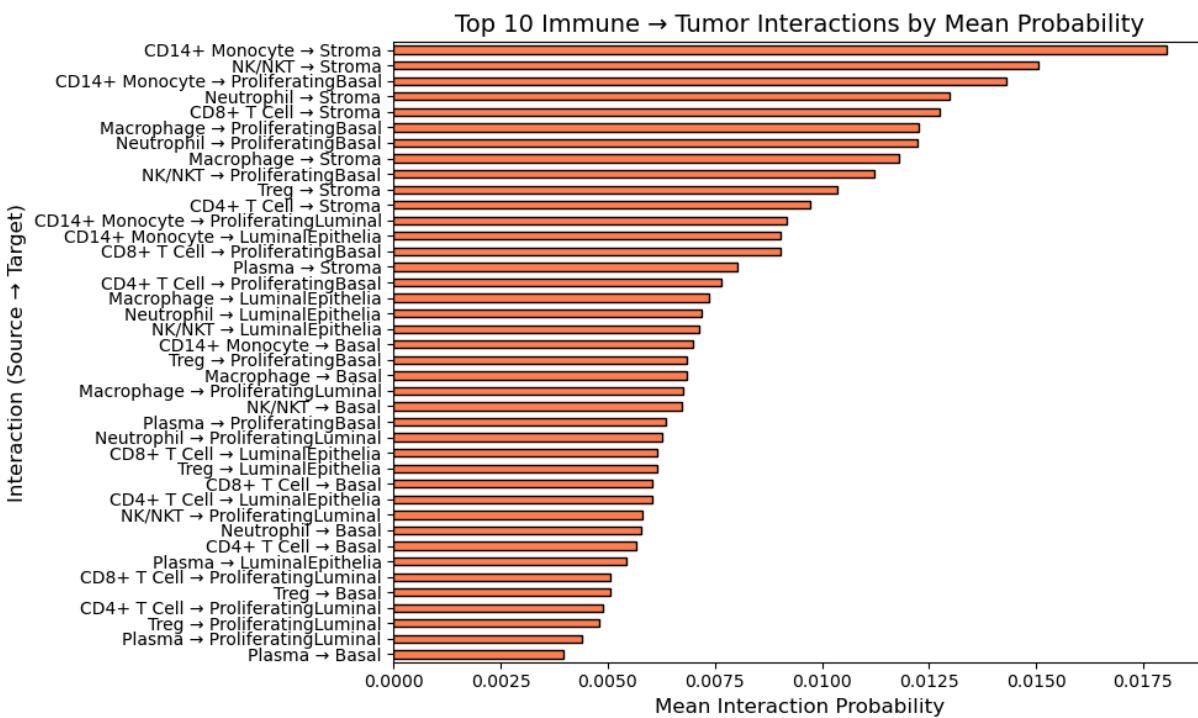
```
plt.tight_layout()
plt.show()
```



```
In [29]: # clean data
edges = data.uns['chat_res'][['source', 'target', 'lr_probs']]
edges = edges.dropna()
edges['lr_probs'] = pd.to_numeric(edges['lr_probs'], errors='coerce')
edges = edges[edges['lr_probs'] > 0] # Keep only positive interaction proba
edges['source'] = edges['source'].astype(str)
edges['target'] = edges['target'].astype(str)

# tumor-immune cells; interaction pairs
tumor_cells = ['LuminalEpithelia', 'Basal', 'Stroma', 'ProliferatingBasal',
immune_cells = ['CD4+ T Cell', 'CD8+ T Cell', 'CD14+ Monocyte', 'NK/NKT', 'N
edges = edges[edges['source'].isin(immune_cells) & edges['target'].isin(tum
edges['interaction'] = edges['source'] + " → " + edges['target']
interaction_scores = edges.groupby('interaction')['lr_probs'].mean().sort_val
top_interactions = interaction_scores.head(50)

# sideways barplot (immune->tumor)
plt.figure(figsize=(10, 6))
top_interactions.sort_values().plot(
    kind='barh', color='coral', edgecolor='black'
)
plt.title("Top 10 Immune → Tumor Interactions by Mean Probability", fontsize=14)
plt.xlabel("Mean Interaction Probability", fontsize=12)
plt.ylabel("Interaction (Source → Target)", fontsize=12)
plt.tight_layout()
plt.show()
```



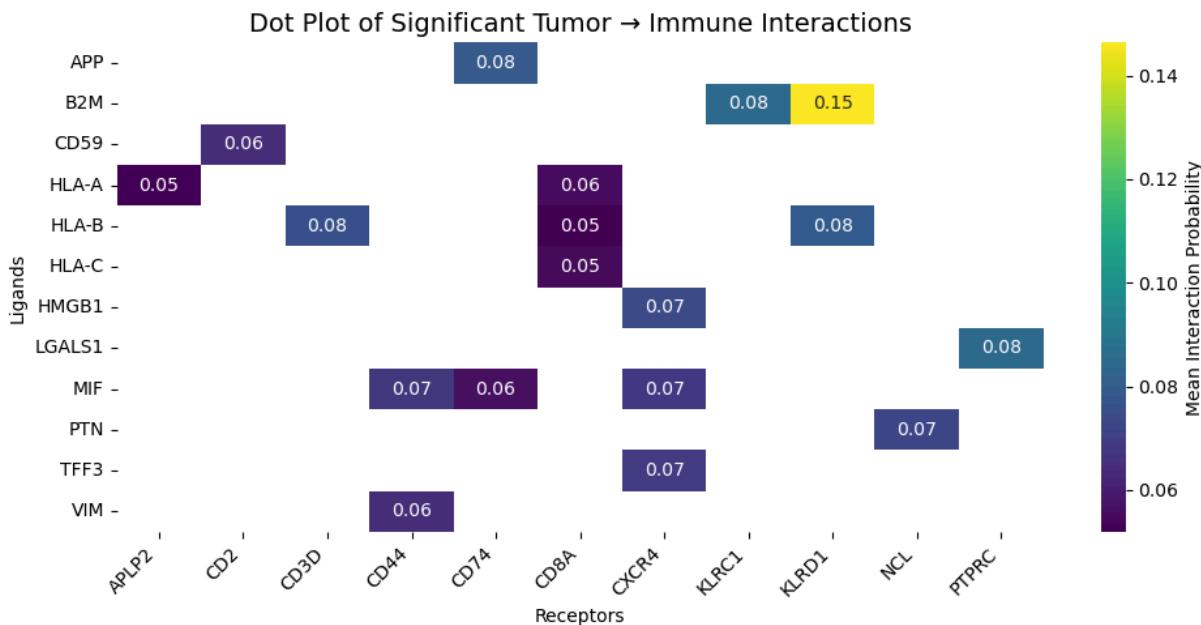
```
In [30]: # clean data
edges = data.uns['chat_res'][['ligand', 'receptor', 'lr_probs', 'cellchat_pvals']]
edges = edges.dropna()
edges['lr_probs'] = pd.to_numeric(edges['lr_probs'], errors='coerce')
edges['cellchat_pvals'] = pd.to_numeric(edges['cellchat_pvals'], errors='coerce')
edges = edges[edges['lr_probs'] > 0]
edges['source'] = edges['source'].astype(str)
edges['target'] = edges['target'].astype(str)

# tumor->immune interactions
tumor_cells = ['LuminalEpithelia', 'Basal', 'Stroma', 'ProliferatingBasal', 'ProliferatingLuminal']
immune_cells = ['CD4+ T Cell', 'CD8+ T Cell', 'CD14+ Monocyte', 'NK/NKT', 'Neutrophil', 'Treg', 'Plasma']

edges = edges[edges['source'].isin(tumor_cells) & edges['target'].isin(immune_cells)]
edges = edges[(edges['lr_probs'] > 0.05) & (edges['cellchat_pvals'] < 0.1)]

# dot plot
dotplot_data = edges.pivot_table(
    index='ligand', columns='receptor', values='lr_probs', aggfunc='mean'
)

plt.figure(figsize=(10, 5))
sns.heatmap(
    dotplot_data, cmap='viridis', annot=True, fmt=".2f", cbar_kws={'label': 'Mean Interaction Probability'}
)
plt.title("Dot Plot of Significant Tumor → Immune Interactions", fontsize=14)
plt.xlabel("Receptors")
plt.ylabel("Ligands")
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```



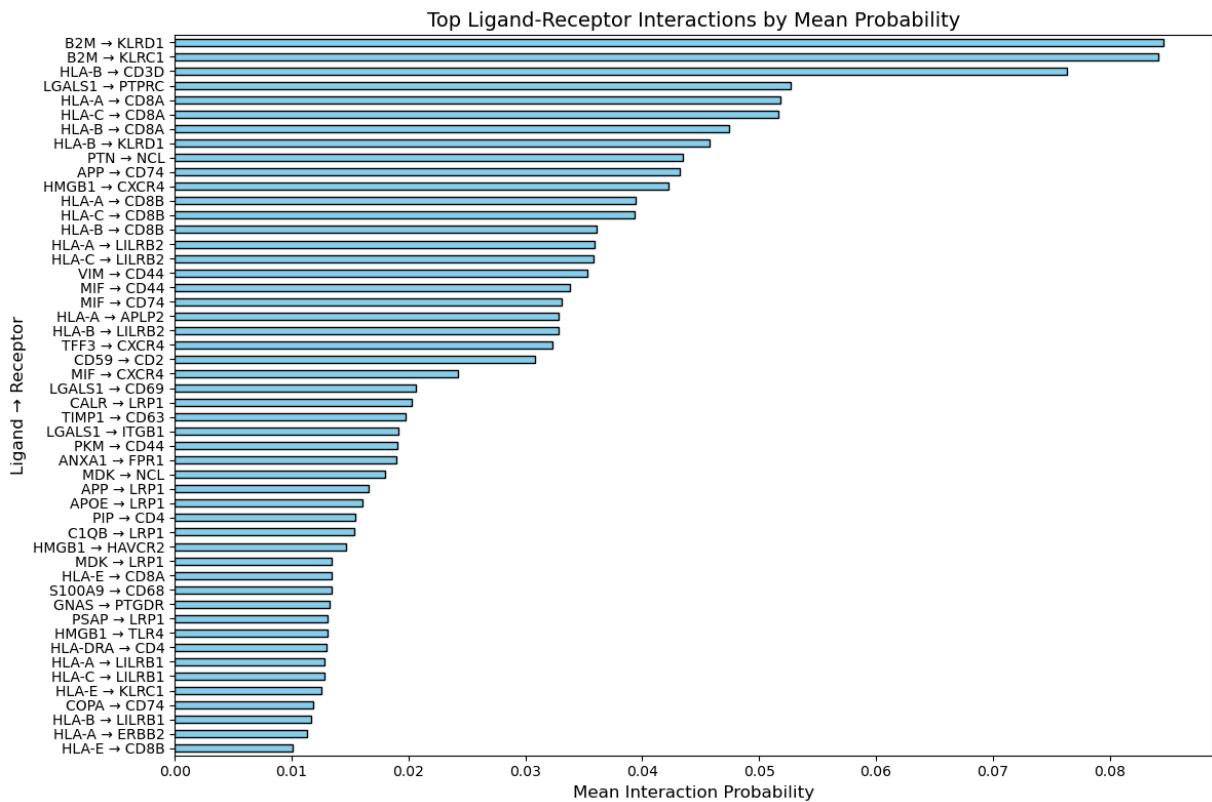
```
In [ ]: # clean data
edges = data.uns['chat_res'][[['ligand', 'receptor', 'lr_probs', 'cellchat_pval', 'cellchat_pval'], 'lr_probs']] = edges.dropna() # Drop rows with missing values
edges['lr_probs'] = pd.to_numeric(edges['lr_probs'], errors='coerce') # Ensure all values are numeric
edges['cellchat_pvals'] = pd.to_numeric(edges['cellchat_pvals'], errors='coerce') # Ensure all values are numeric
edges = edges[edges['lr_probs'] > 0] # Keep only positive interaction probabilities
edges['ligand'] = edges['ligand'].astype(str) # Ensure ligand is a string
edges['receptor'] = edges['receptor'].astype(str) # Ensure receptor is a string

# tumor-immune interactions
tumor_cells = ['LuminalEpithelia', 'Basal', 'Stroma', 'ProliferatingBasal', 'ProliferatingStroma']
immune_cells = ['CD4+ T Cell', 'CD8+ T Cell', 'CD14+ Monocyte', 'NK/NKT', 'Neutrophil', 'RegulatoryT Cell']

edges = edges[edges['source'].isin(tumor_cells) & edges['target'].isin(immune_cells)]
edges['interaction'] = edges['ligand'] + " → " + edges['receptor']

# group by ligand-receptor pairs, calculate mean probabilities, and sort
interaction_scores = edges.groupby('interaction')['lr_probs'].mean().sort_values(ascending=False)
top_interactions = interaction_scores.head(50)

# sideways barplot
plt.figure(figsize=(12, 8))
top_interactions.sort_values().plot(
    kind='barh', color='skyblue', edgecolor='black'
)
plt.title("Top Ligand-Receptor Interactions by Mean Probability", fontsize=14)
plt.xlabel("Mean Interaction Probability", fontsize=12)
plt.ylabel("Ligand → Receptor", fontsize=12)
plt.tight_layout()
plt.show()
```



## Method: NATMI

In [31]:

```
# run natmi
natmi(adata = data,
       groupby='subcluster',
       resource_name='consensus',
       expr_prop=0.1,
       verbose=True, key_added='natmi_res')
```

```
Using resource `consensus`.
Using `_.raw`!
/Users/eunicelee153/fsl/lib/python3.11/site-packages/anndata/_core/anndata.py:401: FutureWarning: The dtype argument is deprecated and will be removed in late 2024.
5442 features of mat are empty, they will be removed.
/Users/eunicelee153/fsl/lib/python3.11/site-packages/liana/method/_pipe_utils/_pre.py:150: ImplicitModificationWarning: Trying to modify attribute `._obs` of view, initializing view as actual.
/Users/eunicelee153/fsl/lib/python3.11/site-packages/liana/method/_pipe_utils/_pre.py:153: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.
0.04 of entities in the resource are missing from the data.
Generating ligand-receptor stats for 45594 samples and 1793 features
```

In [32]:

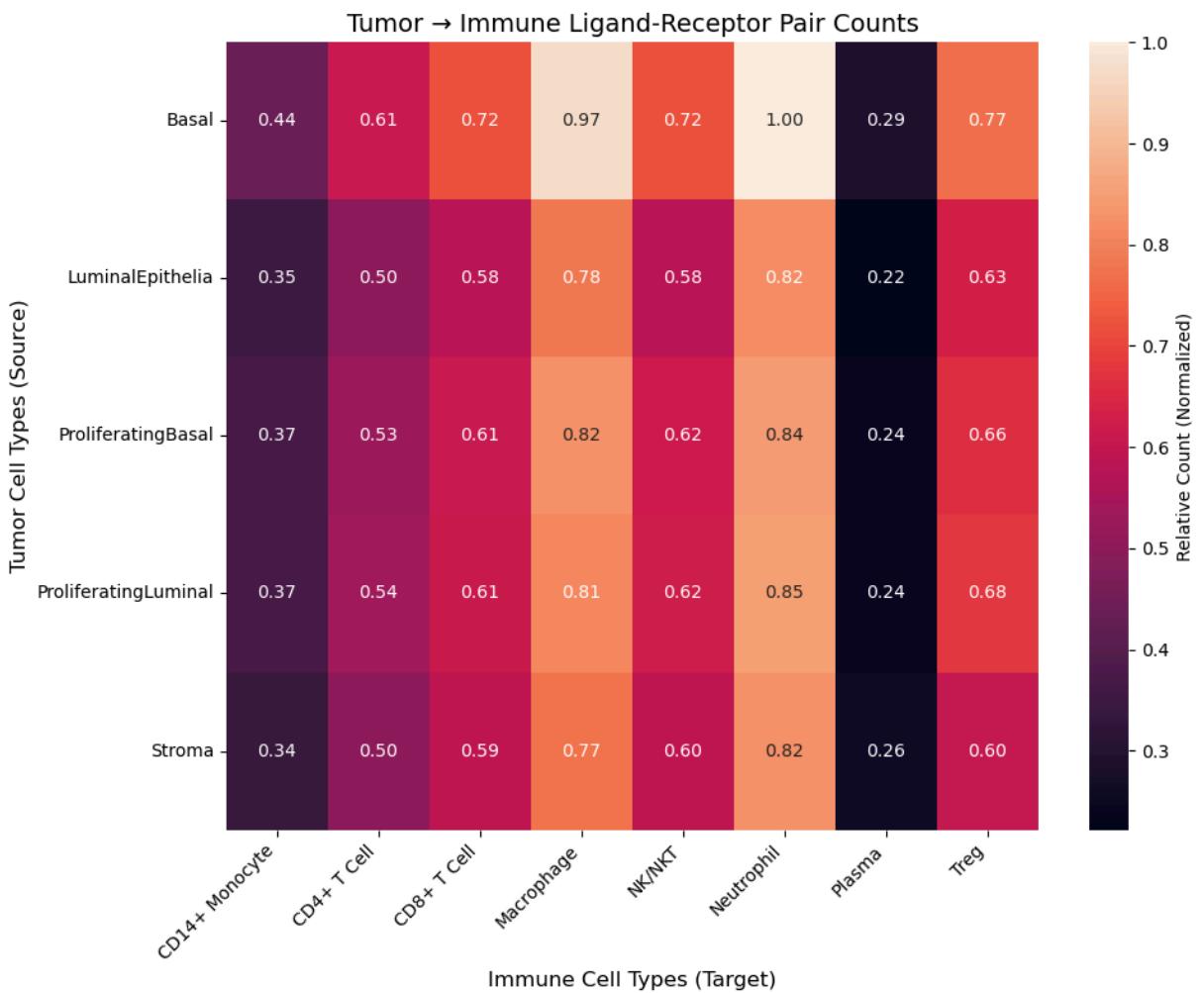
```
# tumor-immune interactions
tumor_cells = ['LuminalEpithelia', 'Basal', 'Stroma', 'ProliferatingBasal',
immune_cells = ['CD4+ T Cell', 'CD8+ T Cell', 'CD14+ Monocyte', 'NK/NKT', 'N']
```

```
# data cleaning
filtered_data = data.uns['natmi_res'][[
    (data.uns['natmi_res']['source'].isin(tumor_cells)) &
    (data.uns['natmi_res']['target'].isin(immune_cells))
]]

heatmap_data = (
    filtered_data
    .groupby(['source', 'target'])
    .size() # Count ligand-receptor pairs for each tumor-immune cell pair
    .unstack(fill_value=0) # Convert to matrix format
)

normalized_data = heatmap_data / heatmap_data.values.max()

# heat map
plt.figure(figsize=(10, 8))
sns.heatmap(
    normalized_data, annot=True, fmt=".2f", cbar_kws={'label': 'Relative Coupling Coefficient'}
)
plt.title("Tumor → Immune Ligand-Receptor Pair Counts", fontsize=14)
plt.xlabel("Immune Cell Types (Target)", fontsize=12)
plt.ylabel("Tumor Cell Types (Source)", fontsize=12)
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```



```
In [36]: # data cleaning
filtered_data = data.ens['natmi_res'][
    (data.ens['natmi_res']['source'].isin(tumor_cells)) &
    (data.ens['natmi_res']['target'].isin(immune_cells))
]

# group by lr pairs and count
gene_level_data = (
    filtered_data.groupby(['ligand', 'receptor']) # Group by gene-level int
    .size() # Count occurrences of each ligand-receptor pair
    .reset_index(name='pair_count') # Reset index to create a new column for
)

heatmap_data = gene_level_data.pivot_table(
    index='ligand', # Rows as ligands
    columns='receptor', # Columns as receptors
    values='pair_count',
    fill_value=0 # Fill missing values with 0
)

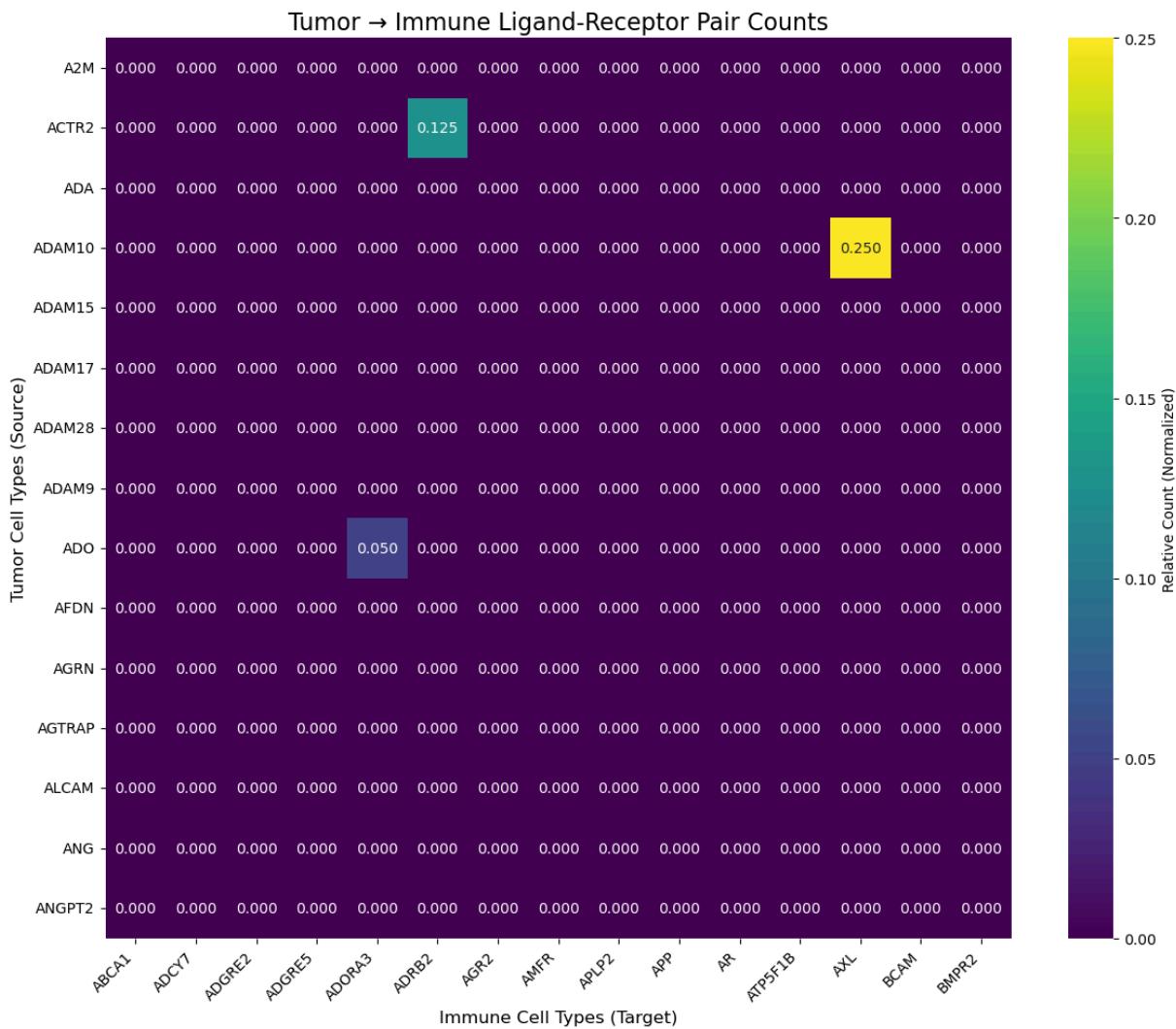
normalized_data = heatmap_data / heatmap_data.values.max()

# heat map
plt.figure(figsize=(12, 10))
sns.heatmap(
```

```

        normalized_data.iloc[:15, :15],
        annot=True,
        fmt=".3f",
        cmap='viridis',
        cbar_kws={'label': 'Relative Count (Normalized)'}
    )
plt.title("Tumor → Immune Ligand-Receptor Pair Counts", fontsize=16)
plt.xlabel("Immune Cell Types (Target)", fontsize=12)
plt.ylabel("Tumor Cell Types (Source)", fontsize=12)
plt.xticks(rotation=45, ha='right')
plt.yticks(rotation=0)
plt.tight_layout()
plt.show()

```



## Method: Rank Aggregate

```

In [37]: # run rank_aggregate
li.mt.rank_aggregate(adata=data,
                      groupby='subcluster',
                      resource_name='consensus',
                      expr_prop=0.1,
                      verbose=True)

```

```

Using resource `consensus`.
Using `.raw`!
/Users/eunicelee153/fsl/lib/python3.11/site-packages/anndata/_core/anndata.py:401: FutureWarning: The dtype argument is deprecated and will be removed in late 2024.
5442 features of mat are empty, they will be removed.
/Users/eunicelee153/fsl/lib/python3.11/site-packages/liana/method/_pipe_utils/_pre.py:150: ImplicitModificationWarning: Trying to modify attribute `._obs` of view, initializing view as actual.
/Users/eunicelee153/fsl/lib/python3.11/site-packages/liana/method/_pipe_utils/_pre.py:153: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.
0.04 of entities in the resource are missing from the data.
Generating ligand-receptor stats for 45594 samples and 1793 features
/Users/eunicelee153/fsl/lib/python3.11/site-packages/liana/method/sc/_liana_pipe.py:262: ImplicitModificationWarning: Setting element `._layers['scaled']` of view, initializing view as actual.
Assuming that counts were `natural` log-normalized!
Running CellPhoneDB
100%|██████████| 1000/1000 [00:33<00:00, 29.95it/s]
Running Connectome
Running log2FC
Running NATMI
Running SingleCellSignalR

```

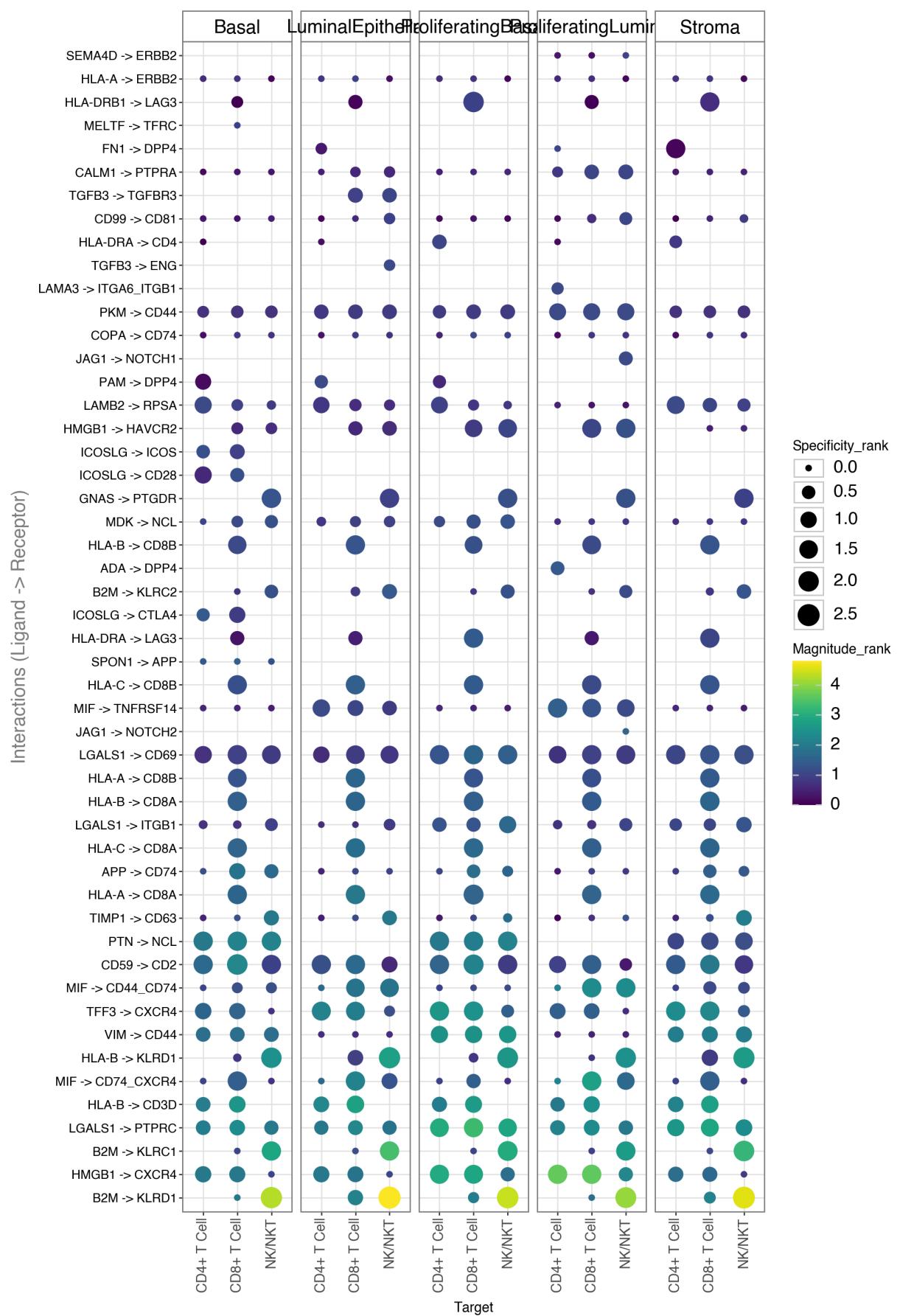
```
In [39]: # dotplot - order by most relevant interactions
li.pl.dotplot(adata = data,
               colour='magnitude_rank',
               size='specificity_rank',
               inverse_size=True,
               inverse_colour=True,
               source_labels=['LuminalEpithelia', 'Basal', 'Stroma', 'Proliferative'],
               target_labels=['CD4+ T Cell', 'CD8+ T Cell', 'NK/NKT'],
               top_n=50,
               orderby='magnitude_rank',
               orderby_ascending=True,
               figure_size=(10, 15)
)
```

```

/Users/eunicelee153/fsl/lib/python3.11/site-packages/liana/plotting/_common.py:104: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

```

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)



In [40]: # dotplot - rank by RRA

```
my_plot = li.pl.dotplot(adata = data,
                        colour='magnitude_rank',
                        inverse_colour=True,
                        size='specificity_rank',
                        inverse_size=True,
                        source_labels=['LuminalEpithelia', 'Basal', 'Stroma'],
                        target_labels=['CD4+ T Cell', 'CD8+ T Cell', 'NK/NKT'],
                        filter_fun=lambda x: x['specificity_rank'] <= 0.01,
                        figure_size=(10, 10))
my_plot
```

