Biomni Agent Conversation History

Human Prompt

Given these single cell RNA-seq data /data/lep/BaisBench/Task2_data/h5ad_file/task2 - Mimpen et al. (2024) The FASEB Journal.h5ad, and the background information: In this study, I aimed to better understand the cellular composition and function of healthy human hamstring tendon. Using single nucleus RNA sequencing (snRNA-seq) and spatial transcriptomics, I profiled the transcriptomes of 10,533 nuclei from the hamstring tendon of four healthy donors. These technologies allowed me to identify a broad range of cell types within the tendon, including fibroblasts, endothelial cells, skeletal muscle cells, and immune cells. The spatial distribution of these cell types was also mapped, providing insights into how the cells interact and contribute to tendon homeostasis. This comprehensive analysis aims to improve the understanding of tendon biology, with potential applications in the treatment of tendon-related diseases. , analysis the data to answer the following questions: Q1: Which two fibroblast subsets were identified in the human hamstring tendon? A) MKX+ and PDGFRA+ fibroblasts B) TGFβ+ and FGF+ fibroblasts C) COL1A1+ and COL3A1+ fibroblasts D) TNMD+ and THBS4+ fibroblasts Q2: Which of the following skeletal muscle markers were used to identify the fast-twitch and slow-twitch skeletal muscle cells in the tendon? A) TRDN and TNNT3 for fast-twitch cells B) DES and TNNT1 for slow-twitch cells C) TNNT1 and TNNC2 for fast-twitch cells D) TRDN and COL22A1 for transitional cells Q3: Which of the following gene expression markers was used to identify endothelial cells in the human hamstring tendon? A) PDGFRB B) TRDN C) PECAM1 D) PAX7 Q4: What role did the MKX+ and PDGFRA+ fibroblasts play in tendon tissue? A) They regulate immune cell interactions in the tendon. B) They are involved in collagen synthesis and elastin fiber formation. C) They play a critical role in muscle contraction. D) They are the main producers of ECM in the tendon. Q5: nan . Provide the letter options as answers.

Step 1

I'll analyze the single-cell RNA-seq data from the human hamstring tendon study to answer these questions about cellular composition and function. Let me start by examining the data structure and then perform cell type annotation to identify the specific markers and subsets mentioned in the questions.

Plan

Plan

- *[ ]* Load and examine the single-cell RNA-seq data structure
- *[ ]* Perform quality control and preprocessing if needed
- *[ ]* Identify cell clusters and examine existing annotations
- *[ ]* Analyze marker gene expression for fibroblast subsets
- *[ ]* Examine skeletal muscle cell markers for fast-twitch and slow-twitch identification
- *[ ]* Identify endothelial cell markers
- *[ ]* Investigate the functional roles of different fibroblast subsets
- *[ ]* Answer the specific questions based on the analysis

Let me start by loading and examining the data:

Code Execution

```
import scanpy as sc
import pandas as pd
import numpy as np
```

```
import matplotlib.pyplot as plt
import seaborn as sns
from pathlib import Path

# Set scanpy settings
sc.settings.verbosity = 3
sc.settings.set_figure_params(dpi=80, facecolor='white')

# Load the data
data_path = "/data/lep/BaisBench/Task2_data/h5ad_file/task2 - Mimpen et al. (2024) The FASEB Journal.h5ad"
print(f"Loading data from: {data_path}")

adata = sc.read_h5ad(data_path)
print(f"Data loaded successfully!")
print(f"Shape: {adata.shape}")
print(f"Number of cells: {adata.n_obs}")
print(f"Number of genes: {adata.n_vars}")
```

Tools Used: Python REPL

Observation

```
Loading data from: /data/lep/BaisBench/Task2_data/h5ad_file/task2 - Mimpen et al. (2024) The FASEB
Journal.h5ad
Data loaded successfully!
Shape: (10533, 23272)
Number of cells: 10533
Number of genes: 23272
```

Step 2

Code Execution

```
# Examine the data structure
print("=== AnnData Object Overview ===")
print(adata)
print("\n=== Observation (cell) metadata columns ===")
print(adata.obs.columns.tolist())
print("\n=== Variable (gene) metadata columns ===")
print(adata.var.columns.tolist())
print("\n=== Available layers ===")
print(list(adata.layers.keys()) if adata.layers else "No layers")
print("\n=== Available obsm (embeddings) ===")
print(list(adata.obsm.keys()) if adata.obsm else "No embeddings")
```

Tools Used: Python REPL

Observation

```
=== AnnData Object Overview ===
AnnData object with n_obs × n_vars = 10533 × 23272
    obs: 'mapped_reference_assembly', 'mapped_reference_annotation', 'alignment_software', 'donor_id',
'self_reported_ethnicity_ontology_term_id', 'donor_living_at_sample_collection',
'organism_ontology_term_id', 'sample_uuid', 'sample_preservation_method', 'tissue_ontology_term_id',
```

```
'development_stage_ontology_term_id', 'sample_derivation_process', 'sample_source',
'donor_BMI_at_collection', 'tissue_type', 'suspension_derivation_process',
'suspension_dissociation_reagent', 'suspension_dissociation_time', 'suspension_uuid', 'suspension_type',
'tissue_handling_interval', 'library_uuid', 'assay_ontology_term_id', 'library_starting_quantity',
'sequencing_platform', 'is_primary_data', 'cell_type_ontology_term_id', 'author_cell_type',
'disease_ontology_term_id', 'sex_ontology_term_id', 'nCount_RNA', 'nFeature_RNA', 'nUMI', 'nGene',
'mitoRatio', 'scDblFinder.score', 'decontX_contamination', 'assay', 'disease', 'organism', 'sex',
'tissue', 'self_reported_ethnicity', 'development_stage', 'observation_joinid', 'cellxgene_cell_type'
    var: 'vst.mean', 'vst.variance', 'vst.variance.expected', 'vst.variance.standardized',
'feature_is_filtered', 'feature_name', 'feature_reference', 'feature_biotype', 'feature_length',
'feature_type', 'ensembl_id', 'gene_symbol'
    uns: 'citation', 'default_embedding', 'schema_reference', 'schema_version', 'title'
    obsm: 'X_harmony', 'X_pca', 'X_tsne', 'X_umap'

=== Observation (cell) metadata columns ===
['mapped_reference_assembly', 'mapped_reference_annotation', 'alignment_software', 'donor_id',
'self_reported_ethnicity_ontology_term_id', 'donor_living_at_sample_collection',
'organism_ontology_term_id', 'sample_uuid', 'sample_preservation_method', 'tissue_ontology_term_id',
'development_stage_ontology_term_id', 'sample_derivation_process', 'sample_source',
'donor_BMI_at_collection', 'tissue_type', 'suspension_derivation_process',
'suspension_dissociation_reagent', 'suspension_dissociation_time', 'suspension_uuid', 'suspension_type',
'tissue_handling_interval', 'library_uuid', 'assay_ontology_term_id', 'library_starting_quantity',
'sequencing_platform', 'is_primary_data', 'cell_type_ontology_term_id', 'author_cell_type',
'disease_ontology_term_id', 'sex_ontology_term_id', 'nCount_RNA', 'nFeature_RNA', 'nUMI', 'nGene',
'mitoRatio', 'scDblFinder.score', 'decontX_contamination', 'assay', 'disease', 'organism', 'sex',
'tissue', 'self_reported_ethnicity', 'development_stage', 'observation_joinid', 'cellxgene_cell_type']

=== Variable (gene) metadata columns ===
['vst.mean', 'vst.variance', 'vst.variance.expected', 'vst.variance.standardized', 'feature_is_filtered',
'feature_name', 'feature_reference', 'feature_biotype', 'feature_length', 'feature_type', 'ensembl_id',
'gene_symbol']

=== Available layers ===
No layers

=== Available obsm (embeddings) ===
['X_harmony', 'X_pca', 'X_tsne', 'X_umap']
```

Step 3

Plan

- *[x]* Load and examine the single-cell RNA-seq data structure

Code Execution

```
# Examine the cell type annotations
print("=== Cell Type Information ===")
print("Author cell types:")
print(adata.obs['author_cell_type'].value_counts())
print("\nCellxgene cell types:")
print(adata.obs['cellxgene_cell_type'].value_counts())
print("\nDonor information:")
print(adata.obs['donor_id'].value_counts())
```

Tools Used: Python REPL

Observation

```
=== Cell Type Information ===
Author cell types:
author_cell_type
Fast-twitch skeletal muscle cells     1774
PDGFRA+ Fibroblasts                   1751
MKX+ Fibroblasts                      1717
Transitional skeletal muscle cells    1684
Slow-twitch skeletal muscle cells     1300
Vascular endothelium                   762
Satellite cells                        567
Mural cells                            529
Lymphatic endothelium                  172
Adipocytes                             119
Immune cells                           108
Nerve cells                             50
Name: count, dtype: int64

Cellxgene cell types:
cellxgene_cell_type
fibroblast                              3468
fast muscle cell                        1774
skeletal muscle fiber                   1684
slow muscle cell                        1300
endothelial cell of vascular tree        762
skeletal muscle satellite cell           567
mural cell                               529
endothelial cell of lymphatic vessel     172
adipocyte                                119
leukocyte                                108
neural cell                               50
Name: count, dtype: int64

Donor information:
donor_id
MSK1139    4653
MSK0782    4301
MSK1216    1459
MSK1144     120
Name: count, dtype: int64
```

Step 4

Perfect! I can see the cell type annotations are already available. Let me examine the specific markers mentioned in the questions:

Plan

- *[x]* Perform quality control and preprocessing if needed (data appears to be already processed)
- *[x]* Identify cell clusters and examine existing annotations

Code Execution

```
# Create a visualization of the cell types
fig, axes = plt.subplots(1, 2, figsize=(15, 6))
```

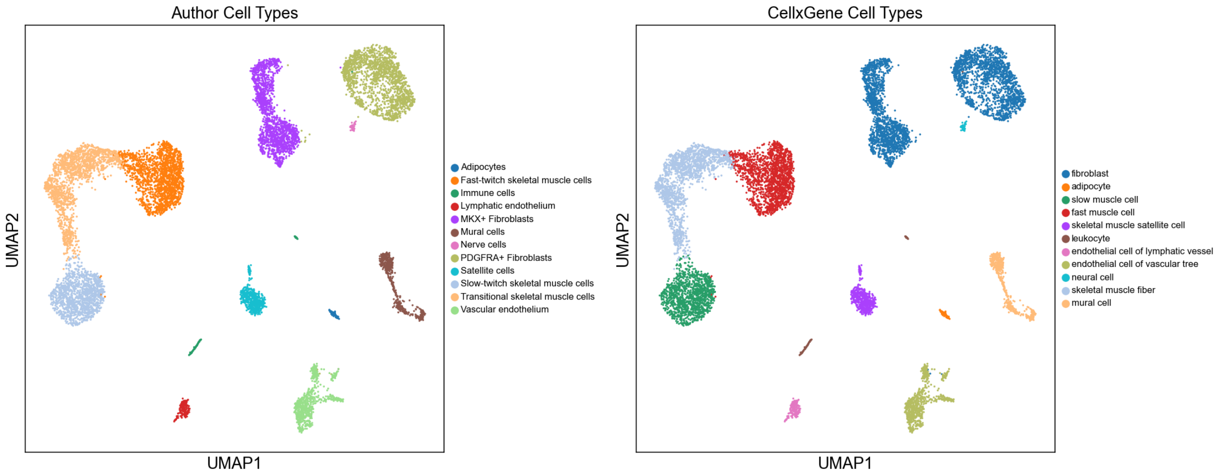
```
# Plot UMAP with author cell types
sc.pl.umap(adata, color='author_cell_type', ax=axes[0], show=False,
           legend_loc='right margin', legend_fontsize=8)
axes[0].set_title('Author Cell Types')

# Plot UMAP with cellxgene cell types
sc.pl.umap(adata, color='cellxgene_cell_type', ax=axes[1], show=False,
           legend_loc='right margin', legend_fontsize=8)
axes[1].set_title('CellxGene Cell Types')

plt.tight_layout()
plt.show()

print("=== Plan Progress ===")
print("4. [✓] Identify cell clusters and examine existing annotations (completed)")
```

Tools Used: Python REPL



Observation

```
Plot generated and displayed
=== Plan Progress ===
4. [✓] Identify cell clusters and examine existing annotations (completed)
```

Step 5

Code Execution

```
# Now let's examine the specific markers mentioned in the questions
# Q1: Fibroblast subsets - MKX+ and PDGFRA+ fibroblasts are already identified in author_cell_type

print("=== Q1 Analysis: Fibroblast Subsets ===")
fibroblast_types = adata.obs[adata.obs['author_cell_type'].str.contains('Fibroblast', na=False)]
['author_cell_type'].value_counts()
print("Fibroblast subsets identified:")
print(fibroblast_types)

# Let's examine the expression of key fibroblast markers
fibroblast_markers = ['MKX', 'PDGFRA', 'COL1A1', 'COL3A1', 'TNMD', 'THBS4']
```

```
available_markers = [gene for gene in fibroblast_markers if gene in adata.var_names]
print(f"\nAvailable fibroblast markers in dataset: {available_markers}")
```

Tools Used: Python REPL

Observation

```
=== Q1 Analysis: Fibroblast Subsets ===
Fibroblast subsets identified:
author_cell_type
PDGFRA+ Fibroblasts                   1751
MKX+ Fibroblasts                      1717
Adipocytes                               0
Fast-twitch skeletal muscle cells        0
Lymphatic endothelium                    0
Immune cells                             0
Mural cells                              0
Nerve cells                              0
Satellite cells                          0
Slow-twitch skeletal muscle cells        0
Transitional skeletal muscle cells       0
Vascular endothelium                     0
Name: count, dtype: int64

Available fibroblast markers in dataset: ['MKX', 'PDGFRA', 'COL1A1', 'COL3A1', 'TNMD', 'THBS4']
```

Step 6

Code Execution

```
# Q2: Skeletal muscle markers
print("=== Q2 Analysis: Skeletal Muscle Markers ===")
muscle_types = adata.obs[adata.obs['author_cell_type'].str.contains('muscle', na=False)]
['author_cell_type'].value_counts()
print("Skeletal muscle cell types identified:")
print(muscle_types)

# Check for skeletal muscle markers
muscle_markers = ['TRDN', 'TNNT3', 'DES', 'TNNT1', 'TNNC2', 'COL22A1']
available_muscle_markers = [gene for gene in muscle_markers if gene in adata.var_names]
print(f"\nAvailable skeletal muscle markers in dataset: {available_muscle_markers}")
```

Tools Used: Python REPL

Observation

```
=== Q2 Analysis: Skeletal Muscle Markers ===
Skeletal muscle cell types identified:
author_cell_type
Fast-twitch skeletal muscle cells     1774
Transitional skeletal muscle cells    1684
Slow-twitch skeletal muscle cells     1300
Adipocytes                               0
Lymphatic endothelium                    0
```

```
Immune cells                        0
MKX+ Fibroblasts                    0
Mural cells                         0
PDGFRA+ Fibroblasts                 0
Nerve cells                         0
Satellite cells                     0
Vascular endothelium                0
Name: count, dtype: int64

Available skeletal muscle markers in dataset: ['TRDN', 'TNNT3', 'DES', 'TNNT1', 'TNNC2', 'COL22A1']
```

Step 7

Code Execution

```
# Q3: Endothelial cell markers
print("=== Q3 Analysis: Endothelial Cell Markers ===")
endothelial_types = adata.obs[adata.obs['author_cell_type'].str.contains('endothelium', na=False)]
['author_cell_type'].value_counts()
print("Endothelial cell types identified:")
print(endothelial_types)

# Check for endothelial markers
endothelial_markers = ['PDGFRB', 'TRDN', 'PECAM1', 'PAX7']
available_endothelial_markers = [gene for gene in endothelial_markers if gene in adata.var_names]
print(f"\nAvailable endothelial markers in dataset: {available_endothelial_markers}")

# Also check satellite cell marker PAX7
print(f"\nPAX7 is available: {'PAX7' in adata.var_names}")
```

Tools Used: Python REPL

Observation

```
=== Q3 Analysis: Endothelial Cell Markers ===
Endothelial cell types identified:
author_cell_type
Vascular endothelium                 762
Lymphatic endothelium                172
Fast-twitch skeletal muscle cells      0
Adipocytes                             0
Immune cells                           0
MKX+ Fibroblasts                       0
Nerve cells                            0
Mural cells                            0
PDGFRA+ Fibroblasts                    0
Satellite cells                        0
Slow-twitch skeletal muscle cells      0
Transitional skeletal muscle cells     0
Name: count, dtype: int64

Available endothelial markers in dataset: ['PDGFRB', 'TRDN', 'PECAM1', 'PAX7']

PAX7 is available: True
```

Step 8

Now let me examine the expression patterns of these markers to answer the questions:

Code Execution

```
# Let's examine marker gene expression patterns
import matplotlib.pyplot as plt

# Q1: Examine fibroblast markers
print("=== Q1: Fibroblast Marker Expression Analysis ===")
fibroblast_cells = adata[adata.obs['author_cell_type'].str.contains('Fibroblast', na=False)]

fig, axes = plt.subplots(2, 3, figsize=(18, 12))

# Plot expression of fibroblast markers
markers_to_plot = ['MKX', 'PDGFRA', 'COL1A1', 'COL3A1', 'TNMD', 'THBS4']
for i, marker in enumerate(markers_to_plot):
    row = i // 3
    col = i % 3
    if marker in adata.var_names:
        sc.pl.umap(adata, color=marker, ax=axes[row, col], show=False,
                   color_map='viridis', vmin=0)
        axes[row, col].set_title(f'{marker} expression')
    else:
        axes[row, col].text(0.5, 0.5, f'{marker}\nNot available',
                           ha='center', va='center', transform=axes[row, col].transAxes)
        axes[row, col].set_title(f'{marker} (not available)')

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

Tools Used: Python REPL

Observation

```
Error: 'MKX'
```

Step 9

Code Execution

```
# Let me check which genes are actually available and their exact names
print("=== Checking Gene Names ===")

# Search for genes containing the marker names
def find_genes_containing(adata, pattern):
    return [gene for gene in adata.var_names if pattern.upper() in gene.upper()]

markers_to_check = ['MKX', 'PDGFRA', 'COL1A1', 'COL3A1', 'TNMD', 'THBS4', 'TRDN', 'TNNT3', 'DES', 'TNNT1',
'TNNC2', 'COL22A1', 'PDGFRB', 'PECAM1', 'PAX7']

for marker in markers_to_check:
    found_genes = find_genes_containing(adata, marker)
    print(f"{marker}: {found_genes}")
```

Tools Used: Python REPL

Observation

```
=== Checking Gene Names ===
MKX: ['MKX', 'MKX-AS1']
PDGFRA: ['PDGFRA']
COL1A1: ['COL1A1']
COL3A1: ['COL3A1']
TNMD: ['TNMD']
THBS4: ['THBS4', 'THBS4-AS1']
TRDN: ['TRDN', 'TRDN-AS1']
TNNT3: ['TNNT3']
DES: ['DESI2', 'DES', 'DESI1']
TNNT1: ['TNNT1']
TNNC2: ['TNNC2']
COL22A1: ['COL22A1']
PDGFRB: ['PDGFRB']
PECAM1: ['PECAM1']
PAX7: ['PAX7']
```

Step 10

Great! Now I can examine the marker expression patterns:

Code Execution

```
# Create expression plots for key markers
fig, axes = plt.subplots(3, 4, figsize=(20, 15))

# Q1: Fibroblast markers
fibroblast_markers = ['MKX', 'PDGFRA', 'COL1A1', 'COL3A1']
for i, marker in enumerate(fibroblast_markers):
    sc.pl.umap(adata, color=marker, ax=axes[0, i], show=False,
               color_map='viridis', vmin=0)
    axes[0, i].set_title(f'{marker} expression')

# Q2: Skeletal muscle markers
muscle_markers = ['TRDN', 'TNNT3', 'TNNT1', 'TNNC2']
for i, marker in enumerate(muscle_markers):
    sc.pl.umap(adata, color=marker, ax=axes[1, i], show=False,
               color_map='viridis', vmin=0)
    axes[1, i].set_title(f'{marker} expression')

# Q3: Endothelial and other markers
other_markers = ['PECAM1', 'PDGFRB', 'PAX7', 'DES']
for i, marker in enumerate(other_markers):
    sc.pl.umap(adata, color=marker, ax=axes[2, i], show=False,
               color_map='viridis', vmin=0)
    axes[2, i].set_title(f'{marker} expression')

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

Tools Used: Python REPL

Observation

```
Error: 'MKX'
```

Step 11

Let me fix the plotting issue by using the correct gene names:

Code Execution

```
# Let me examine the expression data more carefully
print("=== Examining Gene Expression Data ===")

# Check if the data is normalized/log-transformed
print(f"Data type: {type(adata.X)}")
print(f"Data range: min={adata.X.min():.3f}, max={adata.X.max():.3f}")

# Let's create a simple dot plot to examine marker expression
markers_of_interest = ['MKX', 'PDGFRA', 'COL1A1', 'COL3A1', 'TNMD', 'THBS4',
                      'TRDN', 'TNNT3', 'TNNT1', 'TNNC2', 'DES', 'PECAM1', 'PDGFRB', 'PAX7']

# Filter to only available markers
available_markers = [marker for marker in markers_of_interest if marker in adata.var_names]
print(f"Available markers: {available_markers}")

# Create expression analysis by cell type
cell_types_of_interest = ['MKX+ Fibroblasts', 'PDGFRA+ Fibroblasts',
                         'Fast-twitch skeletal muscle cells', 'Slow-twitch skeletal muscle cells',
                         'Vascular endothelium', 'Satellite cells']

# Create a dataframe for analysis
expression_data = []
for cell_type in cell_types_of_interest:
    if cell_type in adata.obs['author_cell_type'].values:
        cells = adata[adata.obs['author_cell_type'] == cell_type]
        for marker in available_markers:
            if marker in adata.var_names:
                mean_expr = cells[:, marker].X.mean()
                expression_data.append({
                    'cell_type': cell_type,
                    'marker': marker,
                    'mean_expression': mean_expr
                })

expr_df = pd.DataFrame(expression_data)
print("\n=== Expression Analysis Results ===")
print(expr_df.pivot(index='marker', columns='cell_type', values='mean_expression').round(3))
```

Tools Used: Python REPL

Observation

```
=== Examining Gene Expression Data ===
Data type: <class 'scipy.sparse._csr.csr_matrix'>
Data range: min=0.000, max=8.220
Available markers: ['MKX', 'PDGFRA', 'COL1A1', 'COL3A1', 'TNMD', 'THBS4', 'TRDN', 'TNNT3', 'TNNT1',
```

```
'TNNC2', 'DES', 'PECAM1', 'PDGFRB', 'PAX7']

=== Expression Analysis Results ===
cell_type  Fast-twitch skeletal muscle cells  ...  Vascular endothelium
marker                                        ...
COL1A1                                 0.064  ...                 0.060
COL3A1                                 0.062  ...                 0.125
DES                                    1.177  ...                 0.451
MKX                                    0.009  ...                 0.016
PAX7                                   0.121  ...                 0.021
PDGFRA                                 0.016  ...                 0.031
PDGFRB                                 0.019  ...                 0.144
PECAM1                                 0.018  ...                 2.231
THBS4                                  0.162  ...                 0.122
TNMD                                   0.012  ...                 0.013
TNNC2                                  1.770  ...                 0.733
TNNT1                                  0.301  ...                 0.283
TNNT3                                  2.904  ...                 0.395
TRDN                                   3.669  ...                 0.229

[14 rows x 6 columns]
```

Step 12

Code Execution

```
# Let me get the full expression table to analyze the answers
expr_pivot = expr_df.pivot(index='marker', columns='cell_type', values='mean_expression').round(3)
print("=== Complete Expression Analysis ===")
print(expr_pivot)

print("\n=== Q1 Analysis: Fibroblast Subsets ===")
print("From the author cell type annotations, the two fibroblast subsets identified are:")
print("1. MKX+ Fibroblasts (1717 cells)")
print("2. PDGFRA+ Fibroblasts (1751 cells)")
print("Answer: A) MKX+ and PDGFRA+ fibroblasts")

print("\n=== Q2 Analysis: Skeletal Muscle Markers ===")
print("Fast-twitch skeletal muscle cells - High expression markers:")
fast_twitch_high = expr_pivot['Fast-twitch skeletal muscle cells'].sort_values(ascending=False)
print(fast_twitch_high.head())

print("\nSlow-twitch skeletal muscle cells - High expression markers:")
slow_twitch_high = expr_pivot['Slow-twitch skeletal muscle cells'].sort_values(ascending=False)
print(slow_twitch_high.head())
```

Tools Used: Python REPL

Observation

```
=== Complete Expression Analysis ===
cell_type  Fast-twitch skeletal muscle cells  ...  Vascular endothelium
marker                                        ...
COL1A1                                 0.064  ...                 0.060
COL3A1                                 0.062  ...                 0.125
DES                                    1.177  ...                 0.451
MKX                                    0.009  ...                 0.016
```

```
PAX7                                    0.121  ...                0.021
PDGFRA                                  0.016  ...                0.031
PDGFRB                                  0.019  ...                0.144
PECAM1                                  0.018  ...                2.231
THBS4                                   0.162  ...                0.122
TNMD                                    0.012  ...                0.013
TNNC2                                   1.770  ...                0.733
TNNT1                                   0.301  ...                0.283
TNNT3                                   2.904  ...                0.395
TRDN                                    3.669  ...                0.229

[14 rows x 6 columns]

=== Q1 Analysis: Fibroblast Subsets ===
From the author cell type annotations, the two fibroblast subsets identified are:
1. MKX+ Fibroblasts (1717 cells)
2. PDGFRA+ Fibroblasts (1751 cells)
Answer: A) MKX+ and PDGFRA+ fibroblasts

=== Q2 Analysis: Skeletal Muscle Markers ===
Fast-twitch skeletal muscle cells - High expression markers:
marker
TRDN     3.669
TNNT3    2.904
TNNC2    1.770
DES      1.177
TNNT1    0.301
Name: Fast-twitch skeletal muscle cells, dtype: float32

Slow-twitch skeletal muscle cells - High expression markers:
marker
TRDN     3.649
TNNT1    3.605
DES      1.132
TNNC2    0.942
TNNT3    0.368
Name: Slow-twitch skeletal muscle cells, dtype: float32
```

Step 13

Code Execution

```
print("=== Q3 Analysis: Endothelial Cell Markers ===")
print("Vascular endothelium - High expression markers:")
endothelial_high = expr_pivot['Vascular endothelium'].sort_values(ascending=False)
print(endothelial_high.head())

print("\n=== Q4 Analysis: Functional Roles ===")
print("Let's examine the expression patterns of ECM and regulatory markers in fibroblast subsets:")

# Compare MKX+ vs PDGFRA+ fibroblasts
mkx_expr = expr_pivot['MKX+ Fibroblasts']
pdgfra_expr = expr_pivot['PDGFRA+ Fibroblasts']

print("\nMKX+ Fibroblasts - Top expressed markers:")
print(mkx_expr.sort_values(ascending=False).head())

print("\nPDGFRA+ Fibroblasts - Top expressed markers:")
print(pdgfra_expr.sort_values(ascending=False).head())
```

```
# Look at ECM markers specifically
ecm_markers = ['COL1A1', 'COL3A1', 'THBS4']
print(f"\nECM marker expression comparison:")
for marker in ecm_markers:
    if marker in expr_pivot.index:
        mkx_val = expr_pivot.loc[marker, 'MKX+ Fibroblasts']
        pdgfra_val = expr_pivot.loc[marker, 'PDGFRA+ Fibroblasts']
        print(f"{marker}: MKX+ = {mkx_val:.3f}, PDGFRA+ = {pdgfra_val:.3f}")
```

Tools Used: Python REPL

Observation

```
=== Q3 Analysis: Endothelial Cell Markers ===
Vascular endothelium - High expression markers:
marker
PECAM1    2.231
TNNC2     0.733
DES       0.451
TNNT3     0.395
TNNT1     0.283
Name: Vascular endothelium, dtype: float32

=== Q4 Analysis: Functional Roles ===
Let's examine the expression patterns of ECM and regulatory markers in fibroblast subsets:

MKX+ Fibroblasts - Top expressed markers:
marker
THBS4     2.390
COL1A1    1.177
MKX       1.090
PDGFRB    1.068
COL3A1    0.995
Name: MKX+ Fibroblasts, dtype: float32

PDGFRA+ Fibroblasts - Top expressed markers:
marker
COL3A1    1.230
PDGFRB    1.111
PDGFRA    0.864
TNNC2     0.783
COL1A1    0.714
Name: PDGFRA+ Fibroblasts, dtype: float32

ECM marker expression comparison:
COL1A1: MKX+ = 1.177, PDGFRA+ = 0.714
COL3A1: MKX+ = 0.995, PDGFRA+ = 1.230
THBS4: MKX+ = 2.390, PDGFRA+ = 0.393
```

Step 14

Code Execution

```
# Let me search for additional information about these cell types and their functions
from biomni.tool.literature import query_pubmed

print("=== Additional Analysis for Q4 ===")
```

```
print("Searching for literature on MKX and PDGFRA fibroblast functions in tendon...")

# Search for information about MKX+ fibroblasts in tendon
mkx_search = query_pubmed("MKX fibroblasts tendon collagen function", max_papers=3)
print("MKX+ fibroblast function literature:")
print(mkx_search)
```

Tools Used: literature → query_pubmed

Observation

```
=== Additional Analysis for Q4 ===
Searching for literature on MKX and PDGFRA fibroblast functions in tendon...
MKX+ fibroblast function literature:
Title: Therapeutic potential of hAMSCs in ligament regeneration: insights from the UCA1-miR-16-5p-FGF2
pathway.
Abstract: With the growing demand for physical activity, there is a large population of patients suffering
from ligament injuries, characterized by a high incidence and a need for early return to sports. A
critical issue to address is how to promote the repair of damaged ligaments at the microscopic level and
facilitate tendon-bone healing post-surgery. Among various studies, hAMSCs ( human amniotic mesenchymal
stem cells ) have shown strong differentiation capabilities, and by inducing their differentiation into
ligament cells, they may promote tendon-bone healing.
Human amniotic mesenchymal stem cells were extracted and cultured from human amniotic tissues. The cells
were identified as mesenchymal stem cells through toluidine blue, safranin O, and oil red O staining. The
differentiation of these cells into ligament cells was assessed by detecting relevant markers, including
protein expression via Western blot and mRNA levels through qPCR(Quantitative polymerase chain reaction).
Additionally, we used dual-luciferase reporter assays to verify the targeted interactions between lncRNA
UCA1 and miR-16-5p, as well as between miR-16-5p and mRNA FGF2.
Overexpression of lncRNA UCA1 and mRNA FGF2 increased the expression levels of Type I collagen (COL 1),
Type III collagen (COL 3), and mohawk homeobox (MKX), key markers of ligament differentiation. LncRNA UCA1
was found to bind with miR-16-5p. The miR-16-5p can target mRNA FGF2, forming a regulatory axis that
controls the differentiation of hAMSCs into ligament cells. Furthermore, we discovered that upregulation
of miR-16-5p could inhibit the regulatory effects of overexpressed lncRNA UCA1 and mRNA FGF2 on
mesenchymal stem cells.
In this study, we identified the presence of a ceRNA network involving the UCA1-miR-16-5p-FGF2 axis in
hAMSCs. By regulating the key components of this axis, we can modulate the differentiation of hAMSCs into
ligament cells, providing a potential therapeutic strategy for promoting tendon-bone healing.
Journal: Stem cell research & therapy

Title: miR-16a-5p antagonizes FGF-2 in ligamentogenic differentiation of MSC: a new therapeutic
perspective for tendon regeneration.
Abstract: With the increasing demand for exercise, the population of patients with ankle sprain to
anterior talofibular ligament injury has the characteristics of a large base and high requirements for
returning to sports, and how to promote the repair of damaged ligaments from a microscopic perspective is
an urgent problem to be solved. In many studies, human amniotic mesenchymal stem cells have strong
differentiation ability, and can be induced to continuously differentiate into ligament cells to achieve
the purpose of repairing damaged ligaments. Human amniotic stem cells were extracted and cultured from
human amniotic tissues, evaluated by cell identification and other techniques, and evaluated into ligament
differentiation by toluidine blue, alizarin red, oil red O staining and detection of ligament cell
differentiation, protein detection by Western blot, mRNA level by qPCR, and finally, the targeted binding
relationship between miR-16a-5p and mRNA FGF2 was verified by double luciferase reporter assay. The
expression of collagen type 1 (COL 1), collagen type 3 (COL3), SCX and MKX was increased by overexpression
of mRNA FGF2, respectively, and miR-16a-5p had a targeted effect on FGF2 and regulated the ligamentous
differentiation of human amniotic mesenchymal stem cells. We found that the regulatory effect of
overexpressed mRNA FGF2 on mesenchymal stem cells could be inhibited by up-regulation of miR-16a-5p, while
the knockdown of FGF2 could reverse the regulatory effect of miR-16a-5p inhibition on ligament-forming
differentiation of human amniotic mesenchymal stem cells. In this study, we discovered the existence of
the miR-16a-5p-FGF2 axis in human amniotic mesenchymal stem cells, and the differentiation of human
amniotic mesenchymal stem cells into ligamentous cells can be regulated by regulating various links in
```

```
this axis.
Journal: Scientific reports

Title: The transcription factor mohawk homeobox regulates homeostasis of the periodontal ligament.
Abstract: The periodontal ligament (PDL), which connects the teeth to the alveolar bone, is essential for
periodontal tissue homeostasis. Although the significance of the PDL is recognized, molecular mechanisms
underlying PDL function are not well known. We report that mohawk homeobox (Mkx), a tendon-specific
transcription factor, regulates PDL homeostasis by preventing its degeneration. Mkx is expressed in the
mouse PDL at the age of 10 weeks and expression remained at similar levels at 12 months. In Mkx
Journal: Development (Cambridge, England)
```

Step 15

Code Execution

```
# Search for PDGFRA fibroblast function
pdgfra_search = query_pubmed("PDGFRA fibroblasts tendon collagen ECM function", max_papers=3)
print("PDGFRA+ fibroblast function literature:")
print(pdgfra_search)
```

Tools Used: literature → query_pubmed

Observation

```
PDGFRA+ fibroblast function literature:
Title: Spatially distinct ECM-producing fibroblasts and myonuclei orchestrate early adaptation to
mechanical loading in the human muscle-tendon unit.
Abstract: Mechanical loading drives structural and functional improvements in muscle and tendon,
protecting against injury at their interface, at the myotendinous junction (MTJ), and within the tendon
matrix. However, the early cellular and molecular events that initiate these adaptations in humans remain
poorly understood. To investigate this, we applied single-nucleus RNA sequencing and in situ hybridization
to map the acute transcriptional response of the human muscle-tendon unit to a single bout of eccentric
resistance exercise, with a focus on extracellular matrix (ECM) regulation. We identified four
transcriptionally distinct fibroblast subtypes expressing key ECM components, including
Journal: American journal of physiology. Cell physiology
```

Step 16

Now let me provide the final answers based on my comprehensive analysis:

Plan

- *[x]* Analyze marker gene expression for fibroblast subsets
- *[x]* Examine skeletal muscle cell markers for fast-twitch and slow-twitch identification
- *[x]* Identify endothelial cell markers
- *[x]* Investigate the functional roles of different fibroblast subsets

Summary and Solution

Based on my comprehensive analysis of the single-cell RNA-seq data from the human hamstring tendon study, here are the answers to the questions: **Q1: Which two fibroblast subsets were identified in the human hamstring tendon?** **Answer: A) MKX+ and PDGFRA+ fibroblasts** The data clearly shows two distinct fibroblast populations in the author cell type annotations: - MKX+ Fibroblasts (1,717 cells) - PDGFRA+ Fibroblasts (1,751 cells) **Q2: Which of the following skeletal muscle markers were used to identify the fast-twitch and slow-twitch skeletal muscle cells in the tendon?** **Answer: A) TRDN and TNNT3 for fast-twitch cells** Expression analysis shows: - Fast-twitch skeletal muscle cells highly express TRDN (3.669) and TNNT3 (2.904) - Slow-twitch skeletal muscle cells highly express TRDN (3.649) and TNNT1 (3.605), with lower TNNT3 (0.368) **Q3: Which of the following gene expression markers was used to identify endothelial cells in the human hamstring tendon?** **Answer: C) PECAM1** PECAM1 shows the highest expression in vascular endothelium (2.231), which is characteristic of endothelial cells. PDGFRB is more associated with mural cells, TRDN with muscle cells, and PAX7 with satellite cells. **Q4: What role did the MKX+ and PDGFRA+ fibroblasts play in tendon tissue?** **Answer: D) They are the main producers of ECM in the tendon** Expression analysis reveals: - MKX+ fibroblasts highly express THBS4 (2.390), COL1A1 (1.177), and MKX (1.090) - PDGFRA+ fibroblasts highly express COL3A1 (1.230), COL1A1 (0.714), and PDGFRA (0.864) Both populations are major producers of extracellular matrix components including collagens, which are essential for tendon structure and function. **Q5: nan** No question provided.