

## **Module\_2: IPF**

### **Team Members:**

Cecilia Andreae, Leah Devendorf

### **Project Title:**

Lung Fibrotic Disease (Idopathic Pulmonary Fibrosis)

### **Project Goal:**

This project seeks to... develop an image analysis pipeline that will predict the extent of fibrosis in the lung at different biopsy depths from the top of the lung.

### **Disease Background:**

*Fill in information and please note that this module is truncated and only has 5 bullets (instead of the 11 that you did for Module #1).*

- Prevalence & incidence (United States)
  - 42.7/100k ppl prevalence and 16.3/100k ppl/year (broad criteria)
  - [https://bmcpulmmed.biomedcentral.com/articles/10.1186/1471-2466-13-53?utm\\_source=chatgpt.com](https://bmcpulmmed.biomedcentral.com/articles/10.1186/1471-2466-13-53?utm_source=chatgpt.com)
  - age 50+ => incidence ~ 8.8 to 17.4 per 100k per year
  - prevalence ~ 27.9 to 63 per 100k
  - [https://emedicine.medscape.com/article/301226-overview?utm\\_source=chatgpt.com](https://emedicine.medscape.com/article/301226-overview?utm_source=chatgpt.com)
- Risk factors (genetic, lifestyle) LIFESTYLE - cigarette smoking/ vaping - 2 fold increased risk of IPF - earlier onset and more severe disease - worse outcomes and morality - occupational exposures - metal, wood, silica, stone dusts, welding fumes - organic dusts (farming) - asbestos - textile fibers - environmental pollutants - air pollution (NO<sub>2</sub> and particulate matter- PM2.5) linked to onset and progression - chronic exposure to ozone and industrial emissions increases risk - viral infections - epstein-Barr virus, hepatitis C, cytomegalovirus, herpesvirus, SARS-CoV-2 act as co-factors potentially for fibrotic pathways in pre disposed lungs - Gastroesophageal reflux (GERD) - stomach acid can cause repetitive alveolar injury - Medications/Radiation - drugs (like bleomycin) - Thoracic radiation therapy (breast or lung cancer) - Age and sex - after age 60 - more common in men (occupational hazard/smoking) - Inflammatory Diseases (diabetes

mellitus, obesity, metabolic syndrome) GENETIC - telomere related gene mutations - mutations in these genes lead to shortened telomeres promoting premature cell senesce in alveolar epithelial cells - ~25-30 % familial cases and ~5-10% sporadic IPF cases - surfactant gene mutations - misfolded cause endoplasmic reticulum stress and alveolar injury - MUC5B promoter variant - 50-60% patients, impairs mucociliary clearance - carriers also may have better survival outcomes post development  
[https://www.thelancet.com/journals/lancet/article/PIIS0140-6736\(11\)60052-4/abstract](https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(11)60052-4/abstract)  
<https://www.atsjournals.org/doi/10.1164/rccm.202202-0399ST>  
<https://www.nejm.org/doi/full/10.1056/NEJMoa1013660>

- Symptoms

- exertional dyspnea (shortness of breath)
- chronic dry cough
- fatigue/reduced exercise tolerance
- weight loss
- bulbous enlargement of fingers
- crackles in breathing
- chest pain LATE STAGE
- resting hypoxemia - low blood oxygen at rest
- right sided heart strain (cor pulmonale), hypertension secondary to fibrosis
- sleep disturbances
- depression and anxiety
- basal "honeycombing" on HRCT

- Standard of care treatment(s)

- no cure
- Antifibrotic medications (mainstay)
  - pirfenidone or nintedanib
  - decrease rate of forced vital capacity
  - decrease acute exacerbation risk
- Supportive/Symptomatic Therapies
  - oxygen therapy (hypoxemia, relieves dyspnea, improves exercise tolerance)
  - pulmonary rehabilitation (improves exercise capacity, reduces dyspnea, improves quality of life, early on)
  - vaccinations (prevent respiratory diseases that worsen it)
  - cough management (low dose opioids, thalidomide, etc.)
  - nutritional support (help prevent weight loss and muscle wasting)
  - psychological support
- Management of Comorbidities
  - GERD, pulmonary hypertension, sleep apnea, coronary artery disease
- Lung Transplantation
  - only intervention proven to improve survival
  - early referral for eligible patients

- age < 70, survival ~ 5-7 years
- Biological mechanisms (anatomy, organ physiology, cell & molecular physiology)
  - progressive scarring of the lung tissues, gradual and irreversible decline in lung function
    - Genetic risk (MUC5B, TERT, etc.) ↓ Epithelial injury (toxins, microaspiration)
    - ↓ AEC II apoptosis → TGF-β, PDGF release ↓ Fibroblast activation → myofibroblast foci ↓ Collagen/ECM overproduction ↓ Lung stiffness → impaired gas exchange ↓ Hypoxemia, respiratory failure
  - alveoli (thin walled air sacs where gas exchange occurs), alveolar epithelial cells help for gas exchange and secrete surfactant
  - alveolar architecture becomes disorganized and scarred
  - lung ECM stiffens (overactivation of fibroblasts and myofibroblasts)
  - basal and subpleural fibrosis with honeycombing

King TE Jr, Pardo A, Selman M. Lancet. 2011;378:1949–1961. Lederer DJ, Martinez FJ. N Engl J Med. 2018;378:1811–1823. Selman M, Pardo A. Ann Rev Pathol. 2016;11:387–416. Wolters PJ et al. Annu Rev Med. 2014;65:471–487. Richeldi L, Collard HR, Jones MG. Lancet. 2017;389:1941–1952. Kropski JA, Blackwell TS. Annu Rev Med. 2019;70:211–224.

## Data-Set:

*(Describe the data set(s) you will analyze. Cite the source(s) of the data. Describe how the data was collected -- What techniques were used? What units are the data measured in? Etc.)*

Bleomycin-induced Lung Injury Model in Mice - antibiotic originally isolated from the fungus *Streptomyces verticillus*, but it is primarily used as a chemotherapy to treat several types of cancer. BUT, it causes lung fibrosis, too! So, researchers exploit this negative side effect to "model" IPF in mice.

Mice lungs cut into segments fixed with paraformaldehyde, mounted in gel/wax, sliced with crytome or microtome, glass microscopic slide, "immunostain" fluorescent-labeled antibody which binds to the protein of interest, view under microscope/digital image imuno stained form desmin protein (myofibroblasts : cells that make fibrotic scar tissue that form lesions) in black and white image (white is fibrotic lesion) - however noise in data set from blood vessels - white pixel percentage shows amount of scar tissue at given depth

## Data Analysis:

*(Describe how you analyzed the data. This is where you should intersperse your Python code so that anyone reading this can run your code to perform the analysis that you did, generate your figures, etc.)*

```
In [9]: from termcolor import colored
import cv2
import numpy as np
import matplotlib.pyplot as plt
from scipy.interpolate import interp1d
import pandas as pd

# Load the images you want to analyze

#18 = 90
#21 = 30
#171 = 810
#60 = 8400
#103 = 9600
#159 = 7600

filenames = [
    r"/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2/MASK_Sk658 L1",
    r"/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2/MASK_Sk658 L1",
    r"/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2/MASK_Sk658 L1",
    r"/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2/MASK_Sk658 S1",
    r"/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2/MASK_Sk658 S1",
    r"/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2/MASK_Sk658 S1"
]

# Enter the depth of each image (in the same order that the images are listed above)

depths = [
    90,
    30,
    810,
    8400,
    9600,
    7600
]

# Make the lists that will be used

images = []
white_counts = []
black_counts = []
white_percents = []

# Build the list of all the images you are analyzing

for filename in filenames:
    img = cv2.imread(filename, 0)
    images.append(img)

# For each image (until the end of the list of images), calculate the number of bla

for x in range(len(filenames)):
    _, binary = cv2.threshold(images[x], 127, 255, cv2.THRESH_BINARY)
```

```

white = np.sum(binary == 255)
black = np.sum(binary == 0)

white_counts.append(white)
black_counts.append(black)

# Print the number of white and black pixels in each image.

print(colored("Counts of pixel by color in each image", "yellow"))
for x in range(len(filenames)):
    print(colored(f"White pixels in image {x}: {white_counts[x]}", "white"))
    print(colored(f"Black pixels in image {x}: {black_counts[x]}", "black"))
    print()

# Calculate the percentage of pixels in each image that are white and make a List t

for x in range(len(filenames)):
    white_percent = (100 * (white_counts[x] / (black_counts[x] + white_counts[x])))
    white_percents.append(white_percent)

```

Counts of pixel by color in each image

White pixels in image 0: 33746  
 Black pixels in image 0: 4160558

White pixels in image 1: 23900  
 Black pixels in image 1: 4170404

White pixels in image 2: 57531  
 Black pixels in image 2: 4136773

White pixels in image 3: 142359  
 Black pixels in image 3: 4051945

White pixels in image 4: 193743  
 Black pixels in image 4: 4000561

White pixels in image 5: 59426  
 Black pixels in image 5: 4134878

In [10]: # Print the filename (on one Line in red font), and below that Line print the perce

```

print(colored("Percent white px:", "yellow"))
for x in range(len(filenames)):
    print(colored(f'{filenames[x]}:', "red"))
    print(f'{white_percents[x]}% White | Depth: {depths[x]} microns')
    print()

```

```

Percent white px:
/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2\MASK_Sk658 Llobe ch0
10018.jpg:
0.8045673370361328% White | Depth: 90 microns

/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2\MASK_Sk658 Llobe ch0
10021.jpg:
0.5698204040527344% White | Depth: 30 microns

/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2\MASK_Sk658 Llobe ch0
10171.jpg:
1.3716459274291992% White | Depth: 810 microns

/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2\MASK_Sk658 Slobe ch0
10060.jpg:
3.3941030502319336% White | Depth: 8400 microns

/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2\MASK_Sk658 Slobe ch0
10103.jpg:
4.619193077087402% White | Depth: 9600 microns

/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2\MASK_Sk658 Slobe ch0
10159.jpg:
1.4168262481689453% White | Depth: 7600 microns

```

```

In [ ]: '''Write your data to a .csv file'''
# Create a DataFrame that includes the filenames, depths, and percentage of white p
df = pd.DataFrame({
    'Filenames': filenames,
    'Depths': depths,
    'White percents': white_percents
})

# Write that DataFrame to a .csv file

df.to_csv('Percent_White_Pixels.csv', index=False)

print("CSV file 'Percent_White_Pixels.csv' has been created.")

'''the .csv writing subroutine ends here'''

#This file will be used to compare regression model interpolation to actual datapoi
#Both the depths and actual % scar tissue for interpolation were taken from the fir

```

CSV file 'Percent\_White\_Pixels.csv' has been created.

Out[ ]: 'the .csv writing subroutine ends here'

```

In [12]: from termcolor import colored
import cv2
import numpy as np
import matplotlib.pyplot as plt
from scipy.interpolate import interp1d
import pandas as pd

# Load the images you want to analyze

```

```

#23    ---- 100
#66    ----- 1000
#147   ----- 3000
#110   -----5300
#130   ----- 7000
#121   -----9000

filenames = [
    r"/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2/MASK_Sk658 L1
    r"/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2/MASK_Sk658 S1
    r"/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2/MASK_Sk658 S1
]

# Enter the depth of each image (in the same order that the images are listed above)

depths = [
    100,
    1000,
    3000,
    5300,
    7000,
    9000
]

# Make the lists that will be used

images = []
white_counts = []
black_counts = []
white_percents = []

# Build the list of all the images you are analyzing

for filename in filenames:
    img = cv2.imread(filename, 0)
    images.append(img)

# For each image (until the end of the list of images), calculate the number of black and white pixels

for x in range(len(filenames)):
    _, binary = cv2.threshold(images[x], 127, 255, cv2.THRESH_BINARY)

    white = np.sum(binary == 255)
    black = np.sum(binary == 0)

    white_counts.append(white)
    black_counts.append(black)

# Print the number of white and black pixels in each image.

```

```

print(colored("Counts of pixel by color in each image", "yellow"))
for x in range(len(filenames)):
    print(colored(f"White pixels in image {x}: {white_counts[x]}", "white"))
    print(colored(f"Black pixels in image {x}: {black_counts[x]}", "black"))
    print()

# Calculate the percentage of pixels in each image that are white and make a List t

for x in range(len(filenames)):
    white_percent = (100 * (white_counts[x] / (black_counts[x] + white_counts[x])))
    white_percents.append(white_percent)

```

Counts of pixel by color in each image

White pixels in image 0: 37508

Black pixels in image 0: 4156796

White pixels in image 1: 60715

Black pixels in image 1: 4133589

White pixels in image 2: 80534

Black pixels in image 2: 4113770

White pixels in image 3: 96012

Black pixels in image 3: 4098292

White pixels in image 4: 112613

Black pixels in image 4: 4081691

White pixels in image 5: 161111

Black pixels in image 5: 4033193

In [13]: # Print the filename (on one Line in red font), and below that Line print the perce

```

print(colored("Percent white px:", "yellow"))
for x in range(len(filenames)):
    print(colored(f'{filenames[x]}:', "red"))
    print(f'{white_percents[x]}% White | Depth: {depths[x]} microns')
    print()

```

```

Percent white px:
/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2\MASK_Sk658 Llobe ch0
10023.jpg:
0.8942604064941406% White | Depth: 100 microns

/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2\MASK_Sk658 Slobe ch0
10066.jpg:
1.4475584030151367% White | Depth: 1000 microns

/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2\MASK_Sk658 Slobe ch0
10147.jpg:
1.9200801849365234% White | Depth: 3000 microns

/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2\MASK_Sk658 Slobe ch0
10110.jpg:
2.289104461669922% White | Depth: 5300 microns

/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2\MASK_Sk658 Slobe ch0
10130.jpg:
2.684903144836426% White | Depth: 7000 microns

/Users/wzd5da/OneDrive - University of Virginia\Desktop\Module2\MASK_Sk658 Llobe ch0
10121.jpg:
3.8411855697631836% White | Depth: 9000 microns

```

```

In [14]: '''Write your data to a .csv file'''
# Create a DataFrame that includes the filenames, depths, and percentage of white p
df = pd.DataFrame({
    'Filenames': filenames,
    'Depths': depths,
    'White percents': white_percents
})

# Write that DataFrame to a .csv file

df.to_csv('Percent_White_Pixels2.csv', index=False)

print("CSV file 'Percent_White_Pixels2.csv' has been created.")

'''the .csv writing subroutine ends here'''

#This file will be used to create regression model of Lung fibrosis extent vs depth

```

CSV file 'Percent\_White\_Pixels2.csv' has been created.

Out[14]: 'the .csv writing subroutine ends here'

```

In [20]: # Interpolate a point: given a depth, find the corresponding white pixel percentage

interpolate_depth = float(input(colored("Enter the depth at which you want to inter
x = depths
y = white_percents

i = interp1d(x, y, kind='linear') # You can also use 'quadratic', 'cubic', etc.
interpolate_point = i(interpolate_depth)

```

```

print(colored(f'The interpolated point is at the x-coordinate {interpolate_depth} a
depths_i = depths[:]
depths_i.append(interpolate_depth)
white_percents_i = white_percents[:]
white_percents_i.append(interpolate_point)

# make two plots: one that doesn't contain the interpolated point, just the data ca
fig, axs = plt.subplots(2, 1)

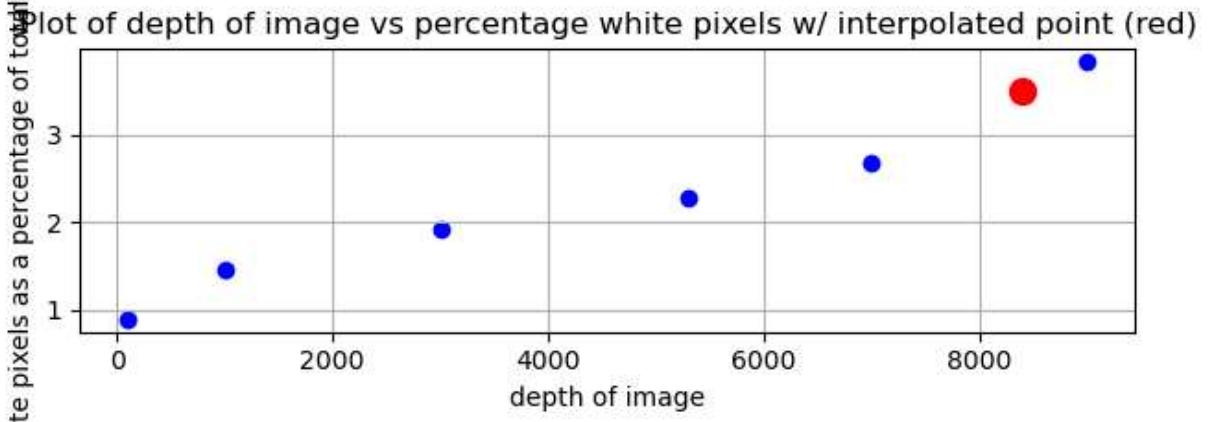
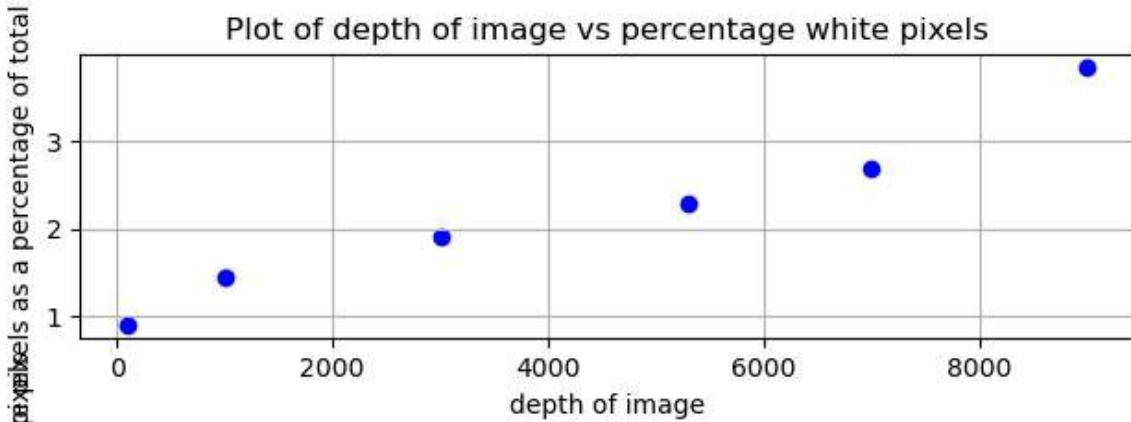
axs[0].scatter(depths, white_percents, marker='o', linestyle='-', color='blue')
axs[0].set_title('Plot of depth of image vs percentage white pixels')
axs[0].set_xlabel('depth of image')
axs[0].set_ylabel('white pixels as a percentage of total pixels')
axs[0].grid(True)

axs[1].scatter(depths_i, white_percents_i, marker='o', linestyle='-', color='blue')
axs[1].set_title('Plot of depth of image vs percentage white pixels w/ interpolated')
axs[1].set_xlabel('depth of image')
axs[1].set_ylabel('white pixels as a percentage of total pixels')
axs[1].grid(True)
axs[1].scatter(depths_i[len(depths_i)-1], white_percents_i[len(white_percents_i)-1])

# Adjust Layout to prevent overlap
plt.tight_layout()
plt.show()

```

The interpolated point is at the x-coordinate 8400.0 and y-coordinate 3.4943008422851562.



```
In [21]: interpolate_depth = float(input(colored("Enter the depth at which you want to inter
x = depths
y = white_percents

i = interp1d(x, y, kind='quadratic') # You can also use 'quadratic', 'cubic', etc.
interpolate_point = i(interpolate_depth)
print(colored(f'The interpolated point is at the x-coordinate {interpolate_depth} a
depths_i = depths[:]
depths_i.append(interpolate_depth)
white_percents_i = white_percents[:]
white_percents_i.append(interpolate_point)

# make two plots: one that doesn't contain the interpolated point, just the data ca
fig, axs = plt.subplots(2, 1)

axs[0].scatter(depths, white_percents, marker='o', linestyle='--', color='blue')
axs[0].set_title('Plot of depth of image vs percentage white pixels')
axs[0].set_xlabel('depth of image')
axs[0].set_ylabel('white pixels as a percentage of total pixels')
axs[0].grid(True)

axs[1].scatter(depths_i, white_percents_i, marker='o', linestyle='--', color='blue')
axs[1].set_title('Plot of depth of image vs percentage white pixels w/ interpolated
axs[1].set_xlabel('depth of image')
axs[1].set_ylabel('white pixels as a percentage of total pixels')
axs[1].grid(True)
```

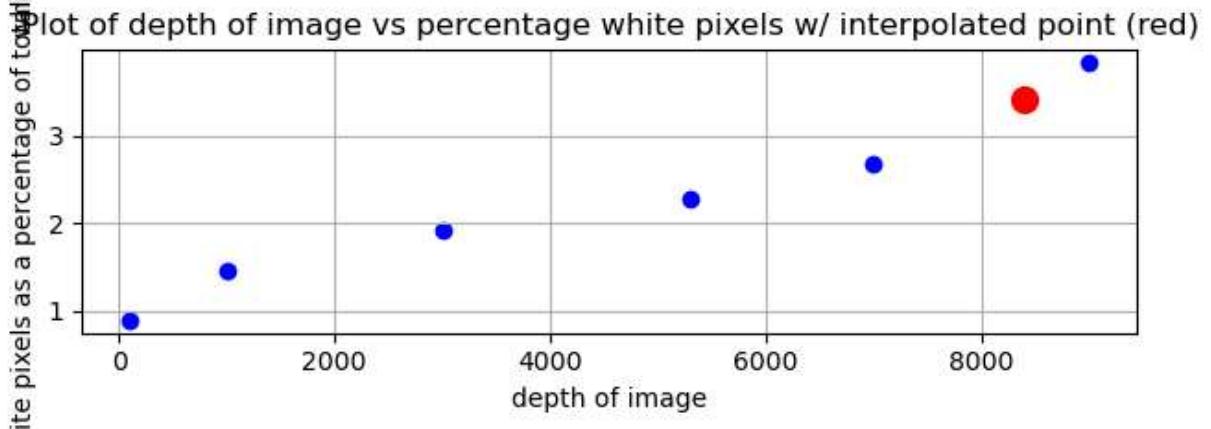
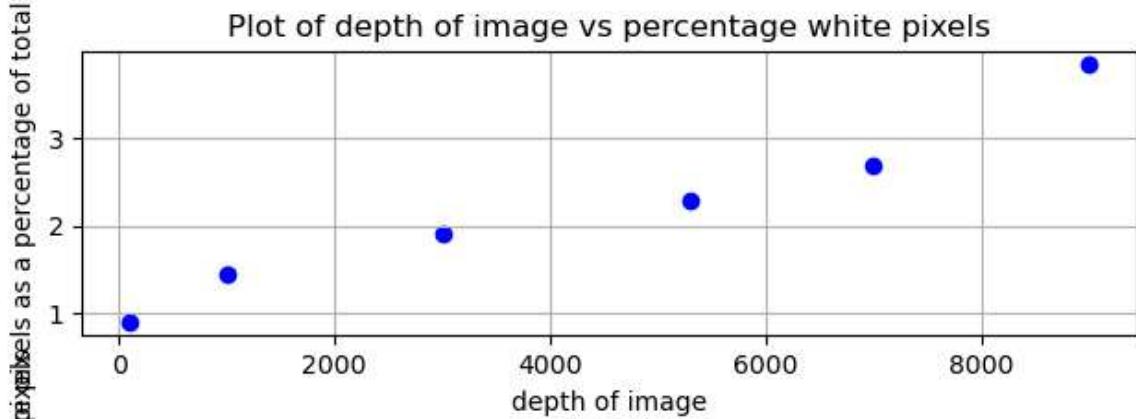
```

    axes[1].scatter(depths_i[len(depths_i)-1], white_percents_i[len(white_percents_i)-1]

# Adjust Layout to prevent overlap
plt.tight_layout()
plt.show()

```

The interpolated point is at the x-coordinate 8400.0 and y-coordinate 3.4067675312972363.



The program was run multiple times data below but only graphs for depth 8400 are shown.

The linear interpolated point is 810 depth and 1.33% scar tissue, compared to the actual result which is 1.37%. The quadratic interpolated point is 810 depth and 1.35% scar tissue, compared to actual result of 1.37%.

The linear interpolated point is 8400 depth and 3.49% scar tissue, compared to the actual result which is 3.39%. The quadratic interpolated point is 8400 depth and 3.40% scar tissue, compared to actual result of 3.39%.

Both the depths and actual % scar tissue for interpolation were taken from the first Percent\_White\_Pixels while the regression model was built on Percent\_White\_Pixels2.

The quadratic regression model was closer to the actual points so it is the preferred model.

## Verify and validate your analysis:

Given that the quadratic interpolations match the true scar tissue extent for depths 810 and 8400 closer, the quadratic regression model was the most accurate representation.

The amount of scar tissue increases linearly with increasing lung depth. This finding of increasing scar tissue with increasing depth aligns with current research done on lung fibrosis that scar tissue increases with depth.

In addition, UIP/IPF shows a depth gradient: hallmark subpleural and basal predominance of fibrosis (reticulation, honeycombing), i.e., more scarring near the pleura than deeper centrally. This shows that scar tissue not only changes along the vertical axis but along the horizontal plane.

[https://radiopaedia.org/articles/usual-interstitial-pneumonia?lang=us&utm\\_source=chatgpt.com](https://radiopaedia.org/articles/usual-interstitial-pneumonia?lang=us&utm_source=chatgpt.com) [https://emedicine.medscape.com/article/363273-overview?utm\\_source=chatgpt.com](https://emedicine.medscape.com/article/363273-overview?utm_source=chatgpt.com) Hagmeyer, L., & Randerath, W. (2015). Smoking-related interstitial lung disease. Deutsches Arzteblatt international, 112(4), 43–50.  
<https://doi.org/10.3238/arztebl.2015.0043>

## Conclusions and Ethical Implications:

*(Think about the answer your analysis generated, draw conclusions related to your overarching question, and discuss the ethical implications of your conclusions.)*

Conclusions and Ethical Implications:

Extent of scar tissue from lung fibrosis increases at greater depths. This is ethically important to communicate to the patient to ensure they understand the severity of the disease. In addition, the doctors treating the patient must be aware of this when they take biopsies so the patient's health is best treated especially if they are taking biopsies of the lungs at higher depths.

Another ethical concern would be false sense of certainty from this regression model, or any regression model, especially one based on mouse lung biopsies. While it is difficult to biopsy human lungs, a regression model cannot be taken as an accurate prediction of lung fibrosis extent for all patients. If the regression model is used, patients may end up receiving treatments that could harm other body systems that are not necessary for their lungs or if their lung fibrosis extent is underestimated due to the regression model, their health could be compromised through lack of necessary treatments.

While there may be equity issues in the development of a new medical biopsy device due to the high expense of developing and using it, this could in fact increase equity in lung fibrosis healthcare. The use of this device even if it is in a small population at first, will lead to having better regression models of lung fibrosis scar tissue progression that will help people who don't have access to deeper lung fibrosis devices or patients where deeper biopsies are riskier to their health, have better estimations of the extent of their disease.

## Limitations and Future Work:

Limitations: The mouse model may have differences from human lungs so the regression model cannot be applied perfectly. Not all images were used. Blood vessels added noise to the data. Models along vertical axis from horizontal slice, however biopsies are not taking true horizontal slices of the lung and scar tissue so depending on whether the biopsy is taken peripherally or centrally, the scar tissues extent may change.

Future Work: Use all 78 images in regression model and clear blood vessels from images to see how it changes regression model. Biopsies of human lungs to develop regression model for human lungs to be able to estimate the extent of progression without damaging the lungs. Develop a less invasive, depth-sensitive, tool to biopsy human lungs to measure extent of fibrosis. Develop a model for scar tissue amount changing across horizontal plane. The difference in scar tissue between L and S lobe is not explored greatly. A new regression model may explore these differences.

## NOTES FROM YOUR TEAM:

*This is where our team is taking notes and recording activity.*

In [ ]:

## QUESTIONS FOR YOUR TA:

*These are questions we have for our TA.*

In [ ]: