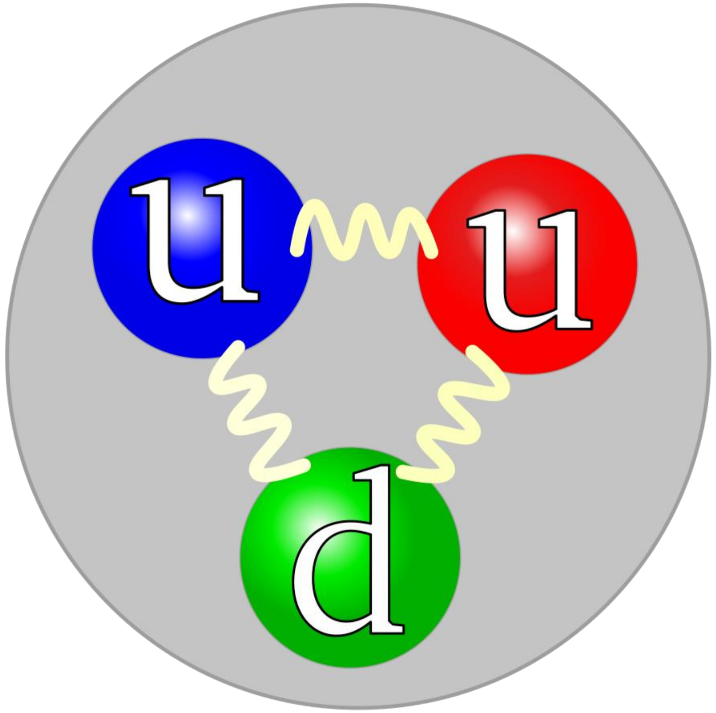
CUA NP Internship: Final Presentation

Leslie Kim

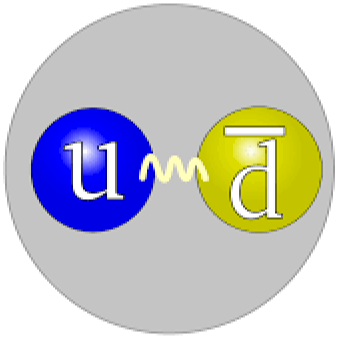
Protons

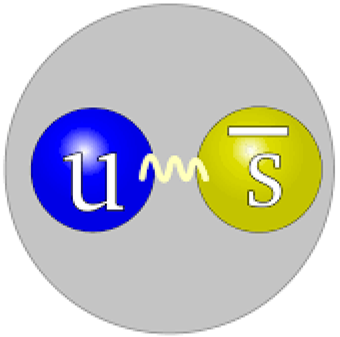
- Made up of 2 up quarks and 1 down quark
- Inelastic Electron Scattering
  - Electrons are accelerated and a beam is shot at a proton target
  - In the collision, pions and kaons are produced
  - They are detected and analyzed by particle detectors
- Studying pions and kaons will help us understand the substructure of proton
  - Pions and kaons are fundamental
  - Can be useful in medical imaging, etc.



Pion & Kaon

- Mesons – made up of one quark and one antiquark
- Pion
  - Made up of an up quark and a down antiquark
- Kaon
  - Made up of an up quark and a strange antiquark
  - Allows us to study the strangeness





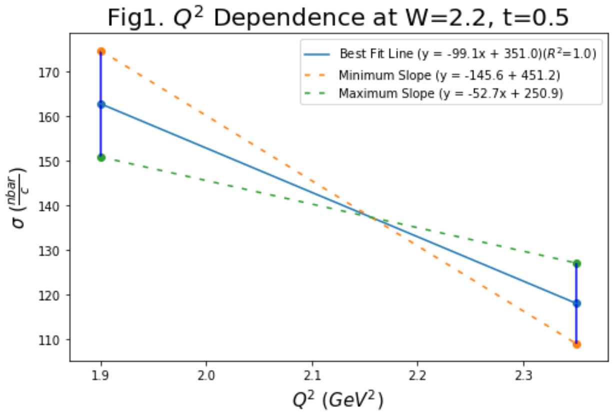
My project

- Global fitting of pion and kaon data
- Python
  - Computer software used for data analysis
  - Used it to plot and fit data
- Fitting Method
  - Linearization – displaying a set of data as a linear fit

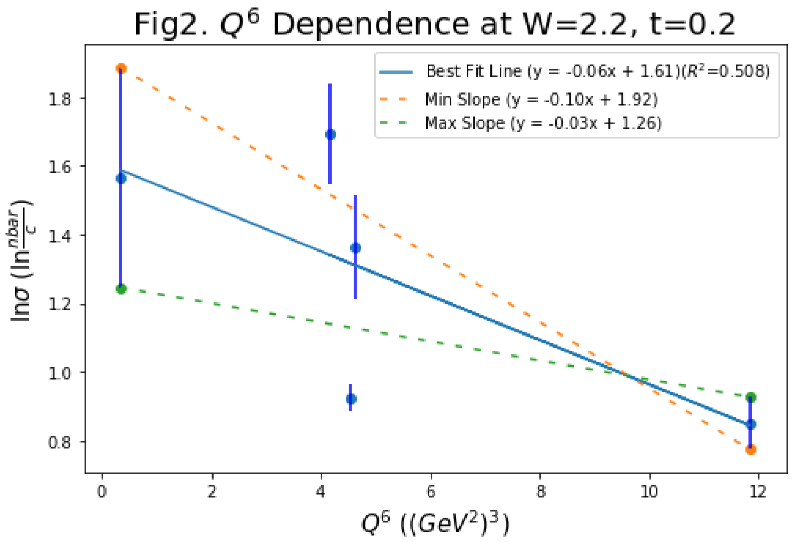
```python
import matplotlib.pyplot as plt
import math
import numpy as np

pts = np.array([(1.90,2.35),(162.7,118.1),(11.9,9.0)])

x = pts[0,:]
y = pts[1,:]
yerror = pts[2,:]

x_min = pts[0,:]
y_min = (pts[1,0]+pts[2,0],pts[1,1]-pts[2,1])
y_min = np.array(y_min)
x_max = pts[0,:]
y_max = (pts[1,0]-pts[2,0],pts[1,1]+pts[2,1])
y_max = np.array(y_max)

m,b = np.polyfit(x,y,1)
m1,b1 = np.polyfit(x_min,y_min,1)
m2,b2 = np.polyfit(x_max,y_max,1)
print(m,b)
print(m1,b1)
print(m2,b2)

correlation_matrix = np.corrcoef(x,y)
correlation_xy = correlation_matrix[0,1]
r_squared = correlation_xy**2
print(r_squared)
```

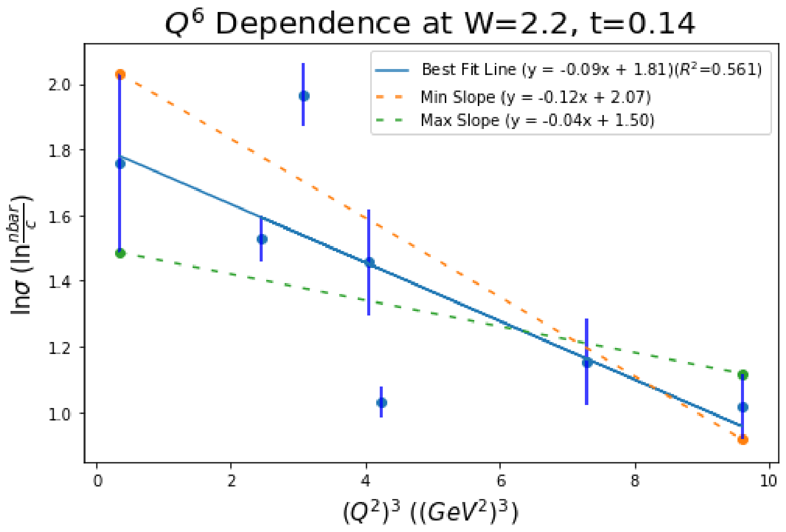
```
-99.111111111111 351.0111111111109
-145.55555555555557 451.1555555555555
-52.666666666666615 250.8666666666665
1.0
```

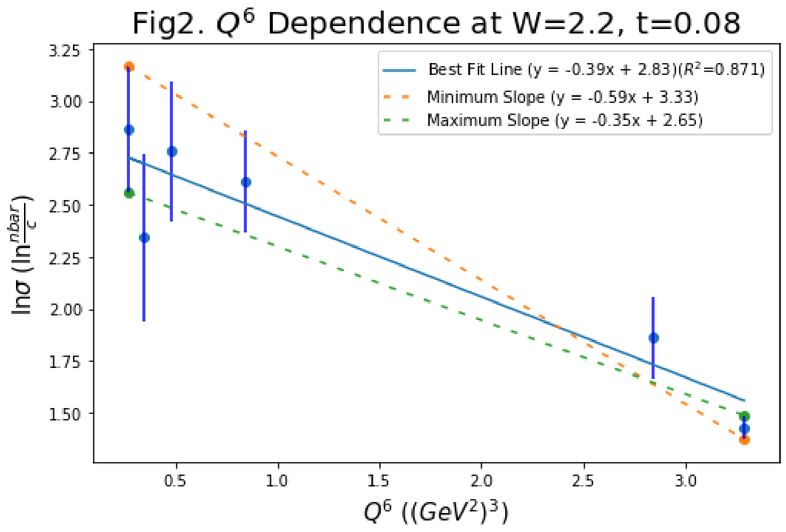
```python
plt.figure(figsize=(8,5))
plt.scatter(x,y)
plt.scatter(x_min,y_min)
plt.scatter(x_max,y_max)
plt.plot(x, m*x + b, label='Best Fit Line (y = -99.1x + 351.0)($R^2$=1.0)')
plt.plot(x_min, m1*x_min + b1, linestyle='--', dashes=(3,5), label='Minimum Slope (y = -145.6 + 451.2)')
plt.plot(x_max, m2*x_max + b2, linestyle='--', dashes=(3,5), label='Maximum Slope (y = -52.7x + 250.9)')
plt.legend()
plt.errorbar(x,y,yerror,fmt=' ', color='blue')
plt.title('Fig1. $Q^2$ Dependence at W=2.2, t=0.5', fontsize =20)
plt.xlabel('$Q^2$ ($GeV^2$)', fontsize =15)
plt.ylabel(r'$\sigma$ ($\frac{nbar}{c}$)', fontsize =15)
plt.show()
```

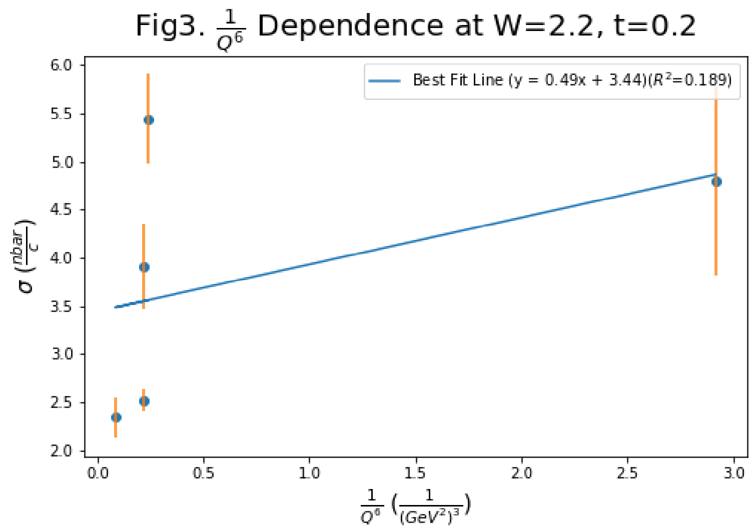


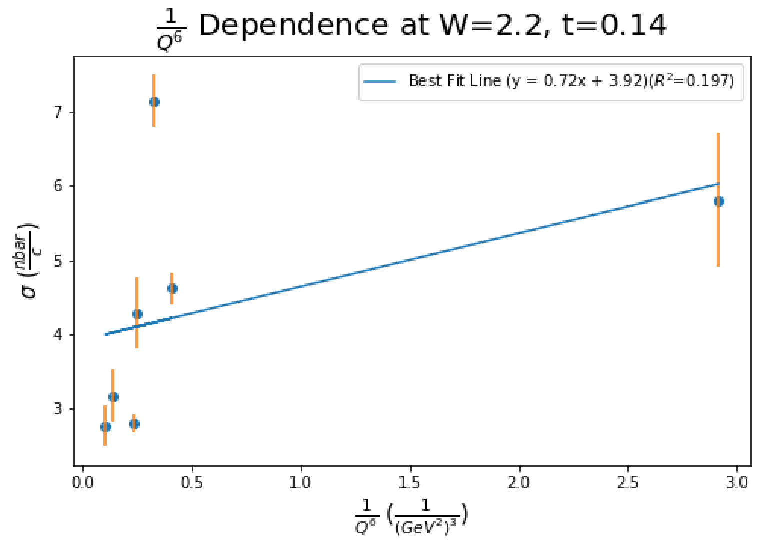
Pion Data Fittings (Q^2)

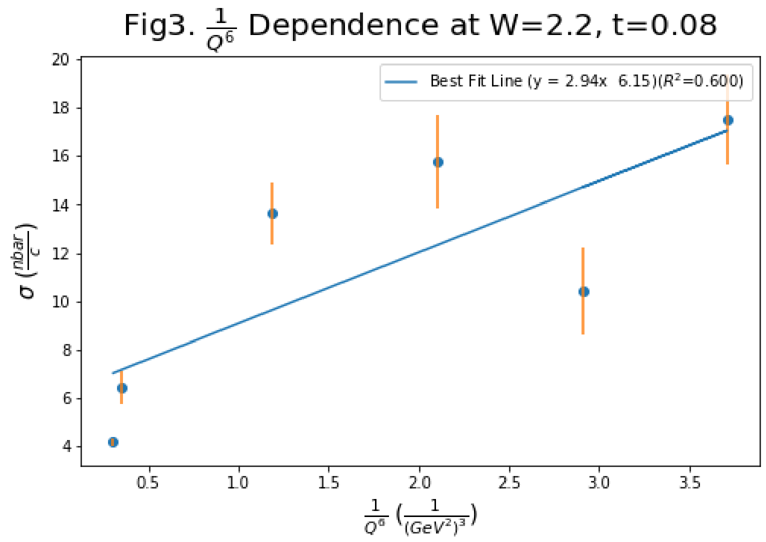












Pion Data Fittings (x)

Fig2. x at Q^2=1.6, W=2.2, t=0.01-0.1, x=0.1-0.2

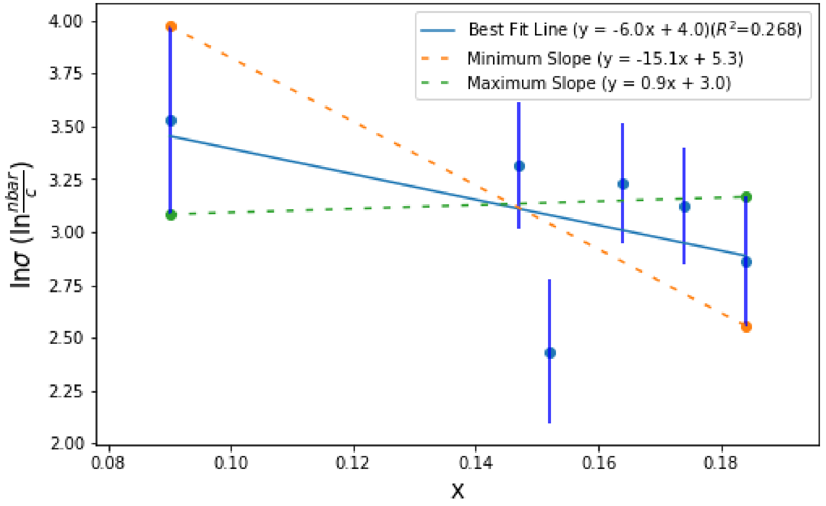


Fig2. $\frac{1}{x^(11)}$ Dependence at Q^2=1.6, W=2.2, t=0.1-0.2, x=0.2-0.3

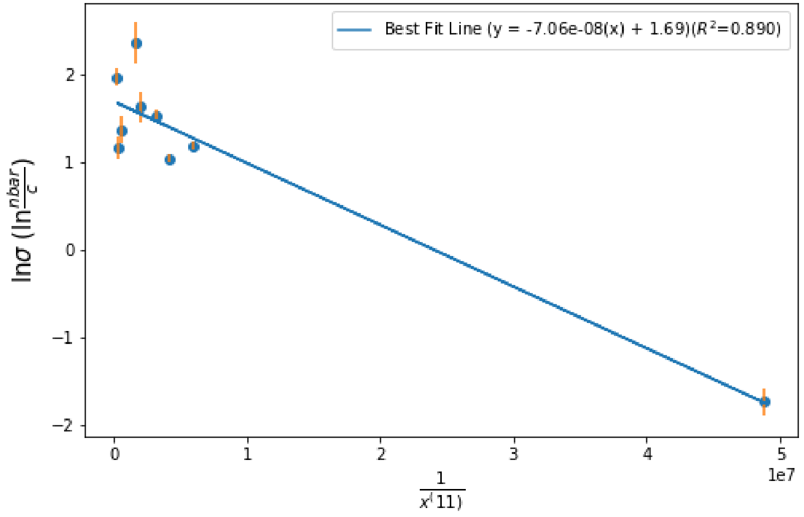
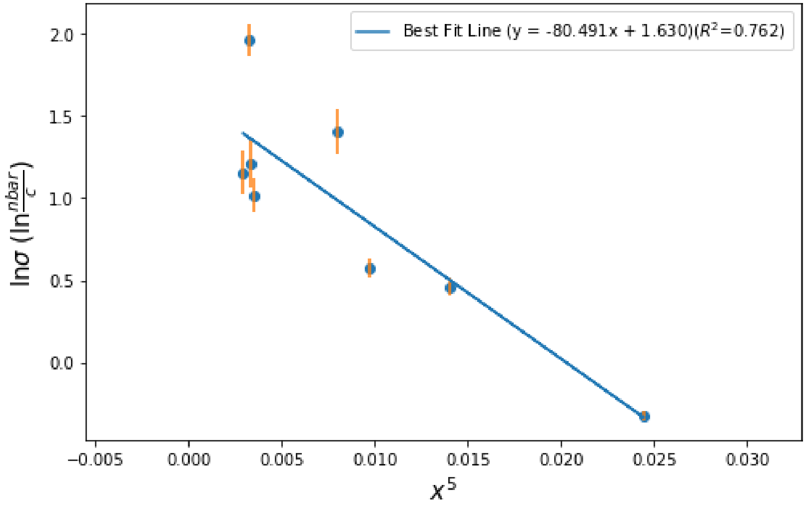
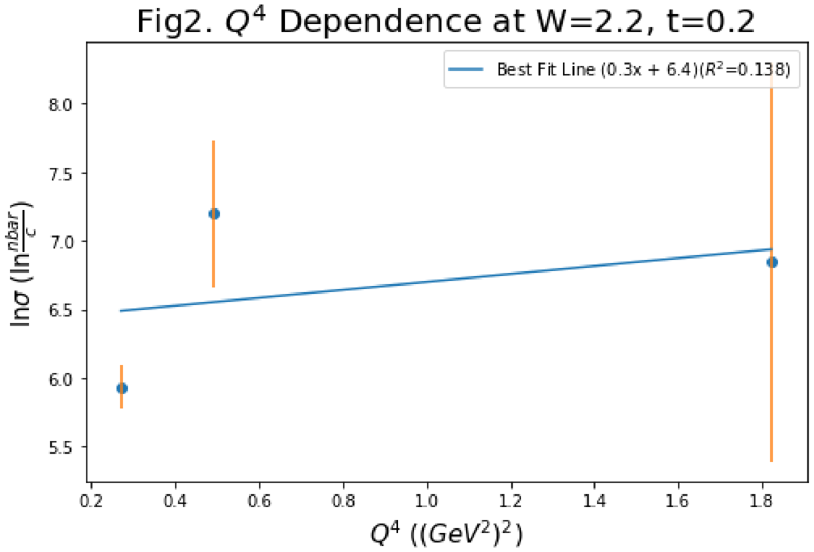
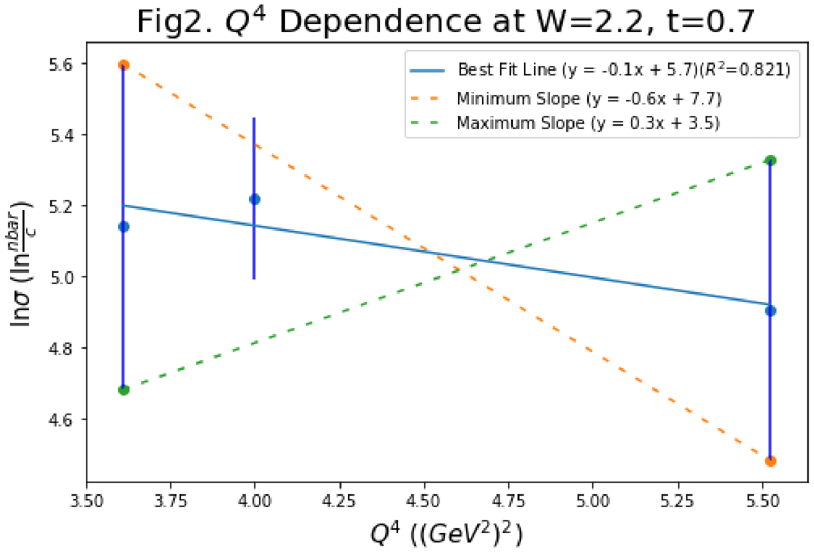


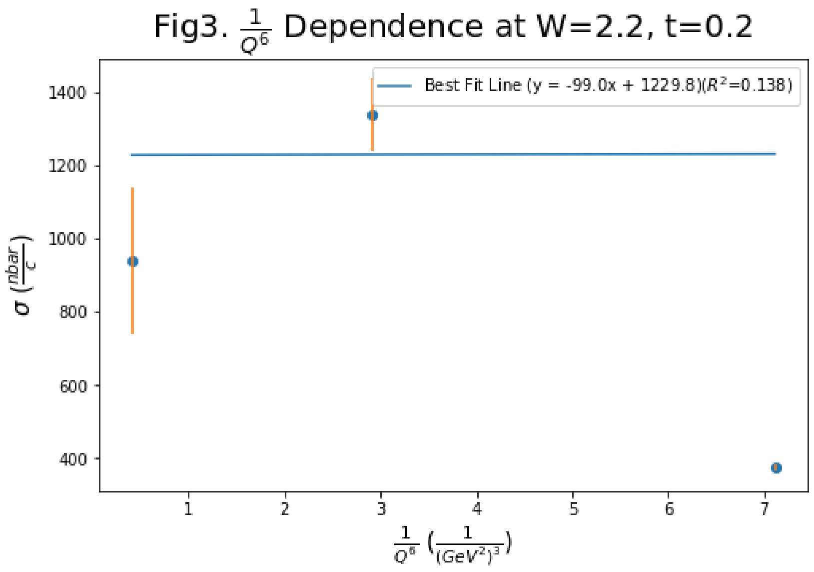
Fig2. x^5 Dependence at Q^2=1.6, W=2.2, t=0.2-0.4, x=0.3-0.5

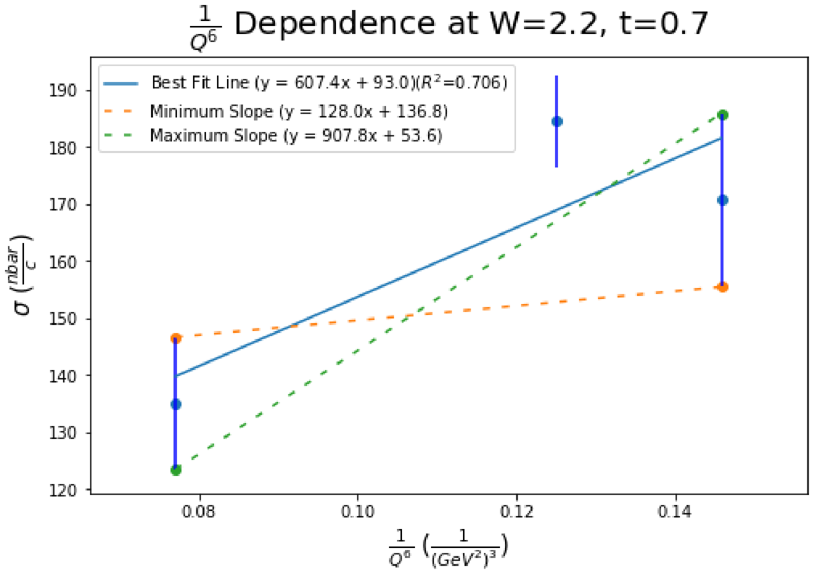


Kaon Data Fittings (Q^2)









Kaon Data Fitting (t)

Fig2. $\frac{1}{t^2}$ Dependence at Q^2=1.6, W=2.2, t=0.2, x=0.1-0.2

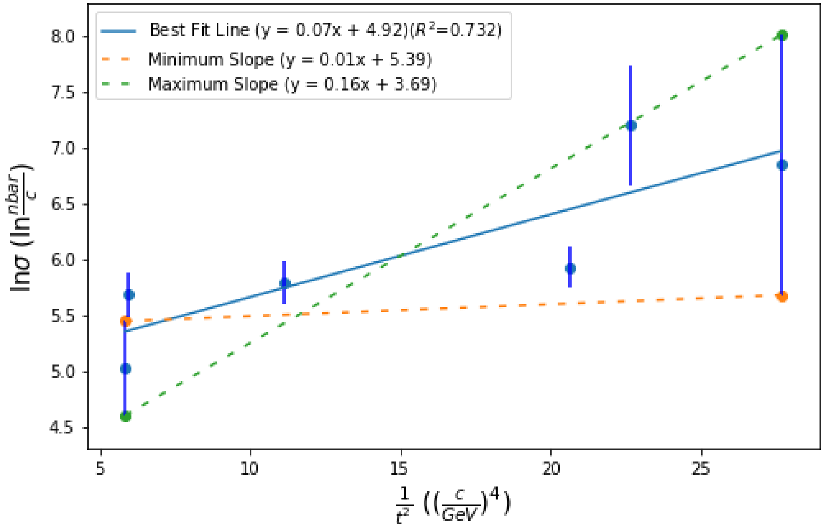


Fig3. $\frac{1}{t^2}$ Dependence (Excluding (0.19,940))

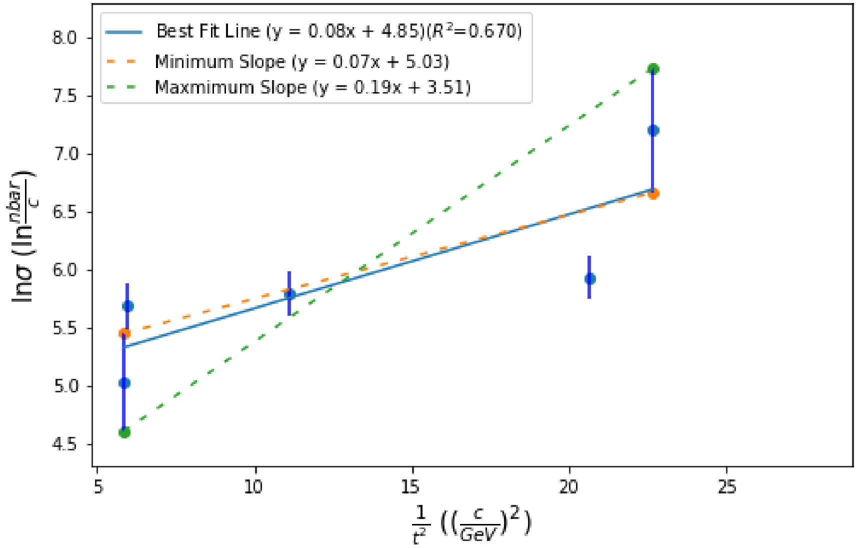


Fig4. $\frac{1}{t^2}$ Dependence (Exluding (0.414,152.1))

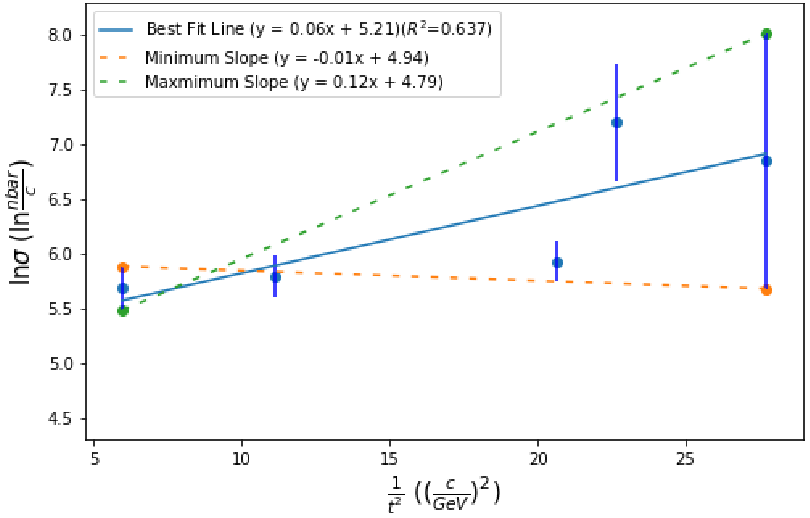
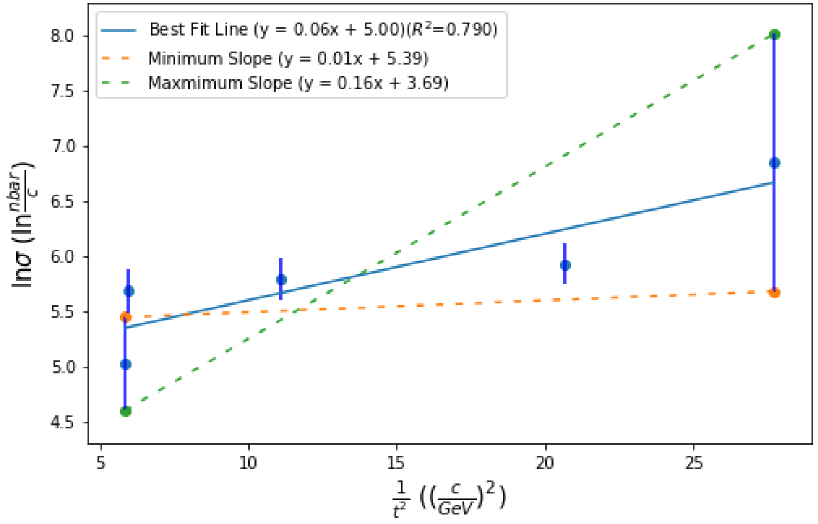
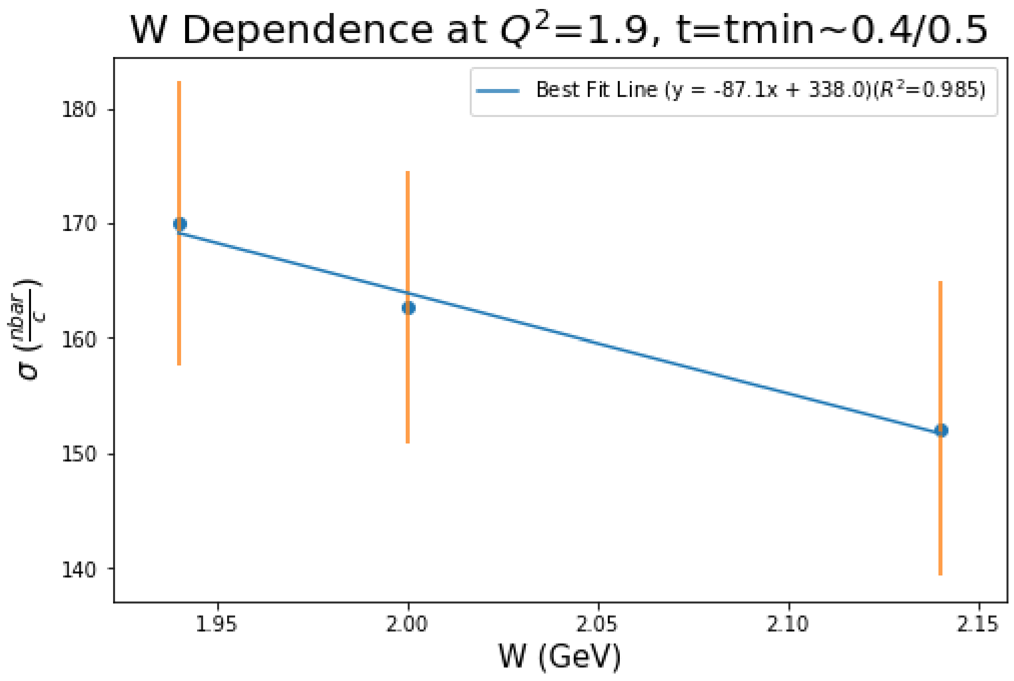
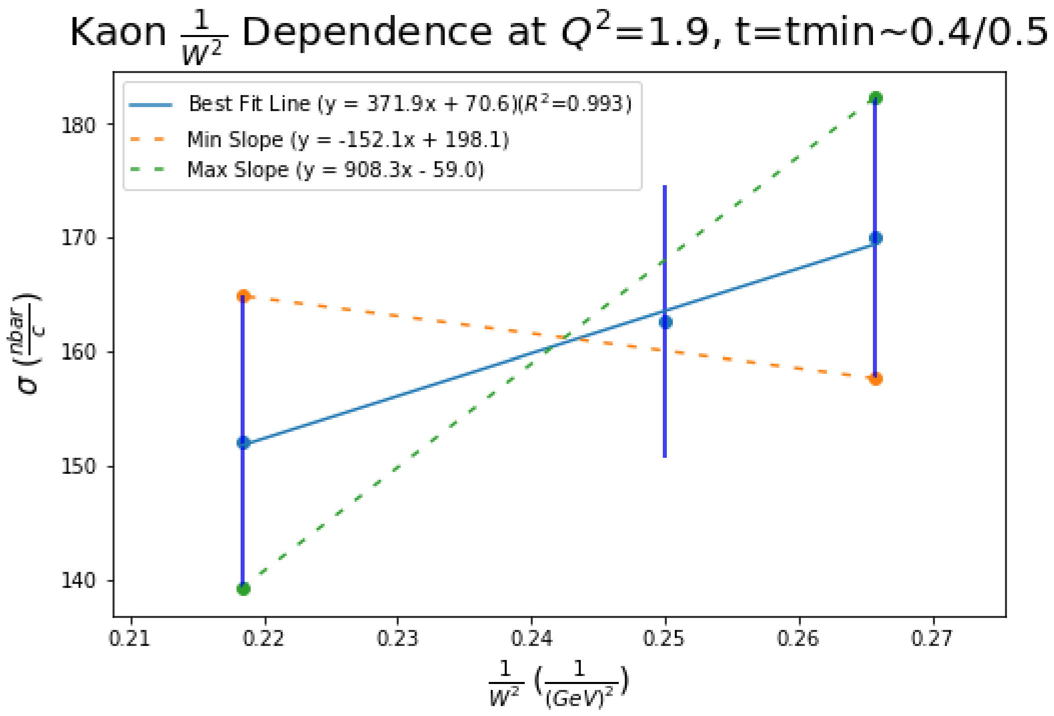


Fig5. $\frac{1}{t^2}$ Dependence (Excluding (0.41,293.9))

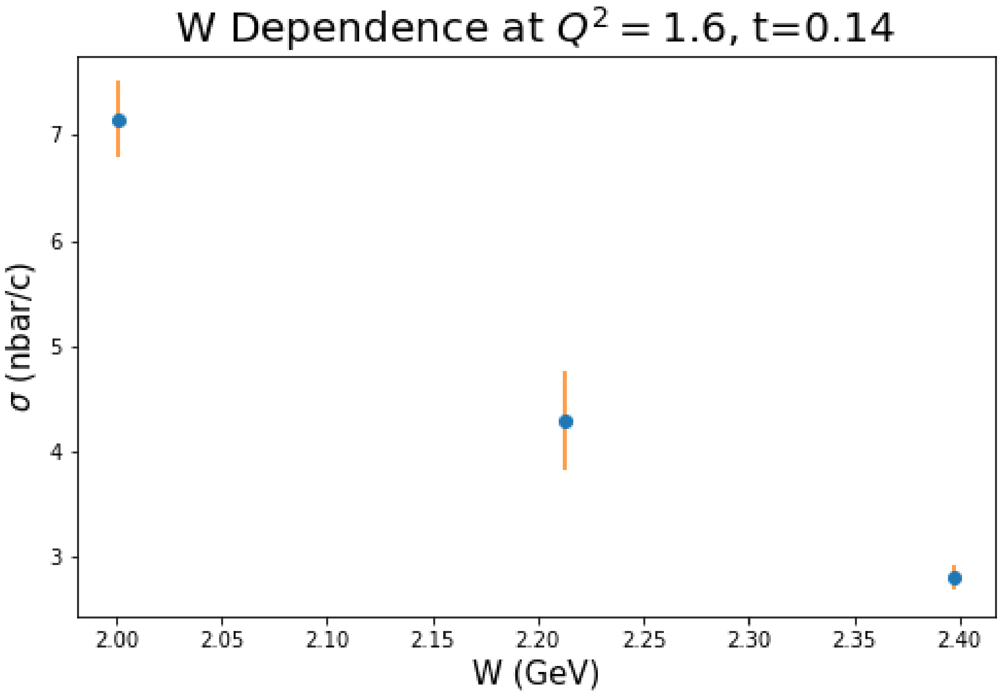


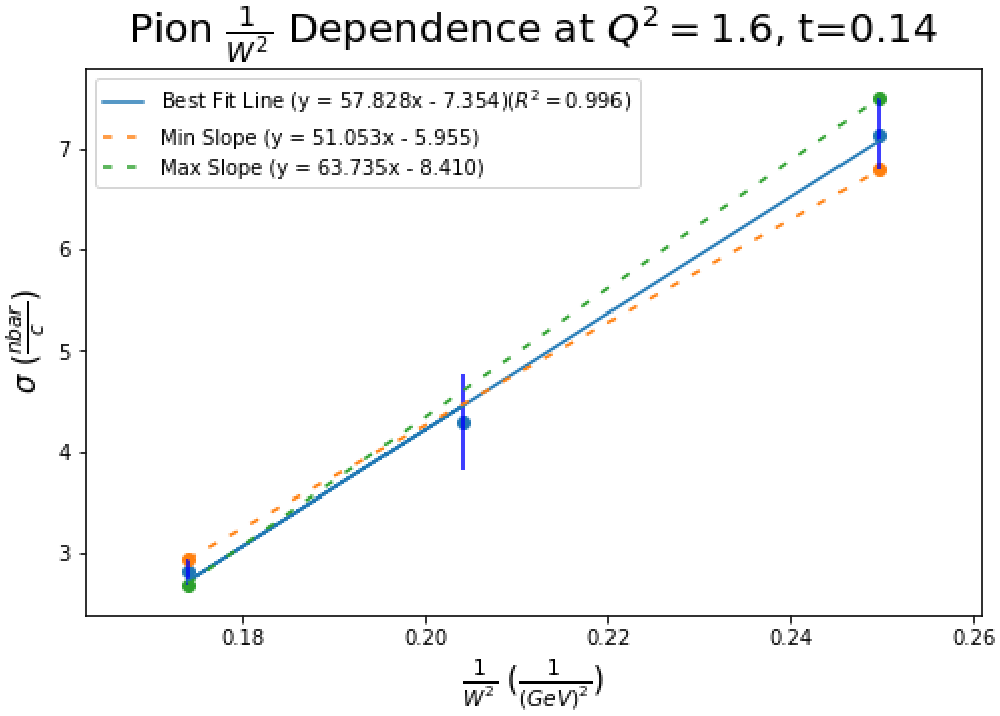
Kaon Data Fitting (W)





Pion Data Fitting (W)





Summary

- Did not get the expected fit for Q^2
  - Q^2 range is limited
  - Larger range and smaller error bars could improve results
- Line of best fit is driven by the points with small error bars
- Pion and kaon fits for W matched up

- Collect more data at a larger range
- Find new data and add them to the existing data set

Thank you!