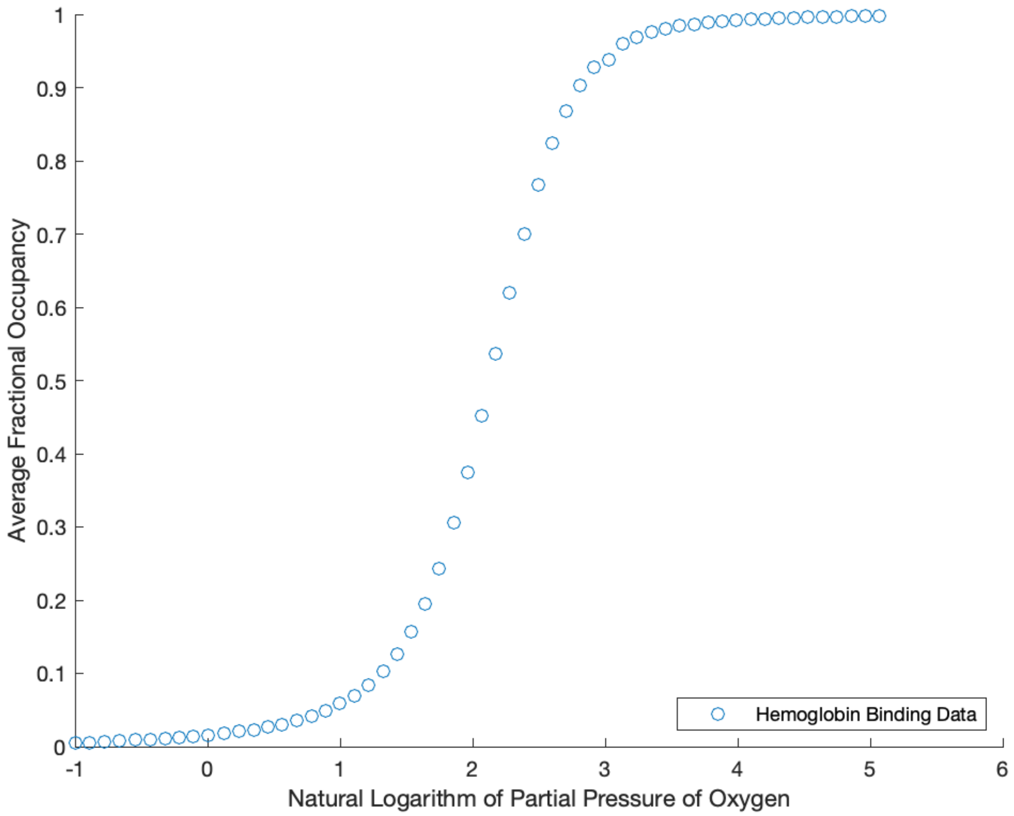
Hemoglobin Cooperation Project

Plotting the Data

In [24]:

```
warning('off','all'); % turn off warnings %
table = readtable("Hemoglobin_Binding_Data.csv");
frac_occupancy = table.FractionalOccupancy;
ln_ppo2 = table.NaturalLogOfPartialPressure;
scatter(ln_ppo2, frac_occupancy)
legend({"Hemoglobin Binding Data"}, "location", "southeast")
xlabel("Natural Logarithm of Partial Pressure of Oxygen")
ylabel("Average Fractional Occupancy")
```

Out[24]:



This is what our graph looks like of the average fractional occupancy vs the natural logarithm of the partial pressure of oxygen. We will attempt to fit our data by tuning some parameters using a couple Models.

The Non-Cooperative Model:

Energy of the system:

$$E = \varepsilon \sum_{\alpha=1}^{4} \sigma_\alpha \tag{1}$$

Average predicted fractional occupancy:

$$\langle N \rangle = \frac{4x + 12x^2 + 12x^3 + 4x^4}{1 + 4x + 6x^2 + 4x^3 + x^4} \tag{2}$$

The Cooperative Model (Pauling):

Energy of the system:

$$E = \varepsilon \sum_{\alpha=1}^{4} \sigma_\alpha + \frac{J}{2} \sum_{(\alpha,\gamma)}{}' \sigma_\alpha \sigma_\gamma \tag{3}$$

Average predicted fractional occupancy:

$$\langle N \rangle = \frac{4x + 12x^2 j + 12x^3 j^3 + 4x^4 j^6}{1 + 4x + 6x^2 j + 4x^3 j^3 + x^4 j^6} \tag{4}$$

The Cooperative Model (Adair):

Energy of the system:

$$E = \varepsilon \sum_{\alpha=1}^{4} \sigma_\alpha + \frac{J}{2} \sum_{(\alpha,\gamma)}{}' \sigma_\alpha \sigma_\gamma + \frac{K}{3!} \sum_{(\alpha,\beta,\gamma)}{}' \sigma_\alpha \sigma_\beta \sigma_\gamma + \frac{L}{4!} \sum_{(\alpha,\beta,\gamma,\delta)}{}' \sigma_\alpha \sigma_\beta \sigma_\gamma \sigma_\delta \tag{5}$$

Average predicted fractional occupancy:

$$\langle N \rangle = \frac{4x + 12x^2 j + 12x^3 j^3 k + 4x^4 j^6 k^4 l}{1 + 4x + 6x^2 j + 4x^3 j^3 k + x^4 j^6 k^4 l} \tag{6}$$

Let's see how these models differ.

The Non-Cooperative Model

Let's assume the model is non-cooperative. Then we can use the following equation to predict the fractional occupancy.

$$\langle N \rangle = \frac{4x + 12x^2 j + 12x^3 j^3 + 4x^4 j^6}{1 + 4x + 6x^2 j + 4x^3 j^3 + x^4 j^6} \tag{7}$$

But since we're assuming non-cooperative, we assume `j=1`, so really the equation looks like:

$$\langle N \rangle = \frac{4x + 12x^2 + 12x^3 + 4x^4}{1 + 4x + 6x^2 + 4x^3 + x^4} \tag{8}$$

First let's reverse the natural log of PPO2 to just PPO2. The reason for doing so will become apparent later.

In [25]:

```
ppo2 = exp(ln_ppo2);
```

To fit `x`, we need to minimize the error of the data between the model. We can do this by introducing a fit parameter, `b`.

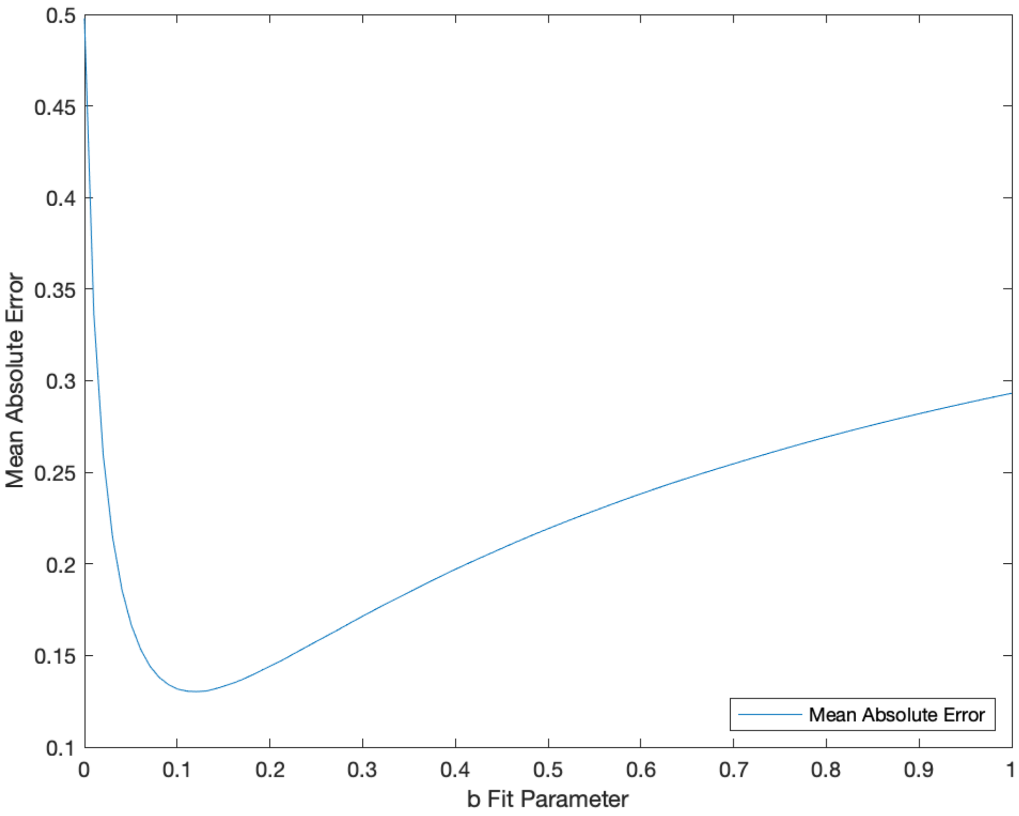
$$x = e^{ln(P_{O_2})}b = P_{O_2}b \tag{9}$$

Let's try 1000 elements between [0, 1) and look at the mean absolute error.

In [26]:

```
b = linspace(0, 1, 100);
x = ppo2.*b;
numerator = 4.*x+12.*x.^2+12.*x.^3+4.*x.^4;
denominator = 1+4.*x+6.*x.^2+4.*x.^3+x.^4;
N_hat = numerator ./ denominator ./ 4;
error = sum(abs(N_hat - frac_occupancy)) ./ length(ln_ppo2);
plot(b, error)
legend({"Mean Absolute Error"}, "location", "southeast")
xlabel("b Fit Parameter")
ylabel("Mean Absolute Error")
```

Out[26]:



We can use the `min` function to find the `b` with the lowest mean absolute error.

In [27]:

```
m = min(error)
```

```
b = b(find(error==m))
```

Out[27]: m =

```
  0.1303
```

Out[27]: b =

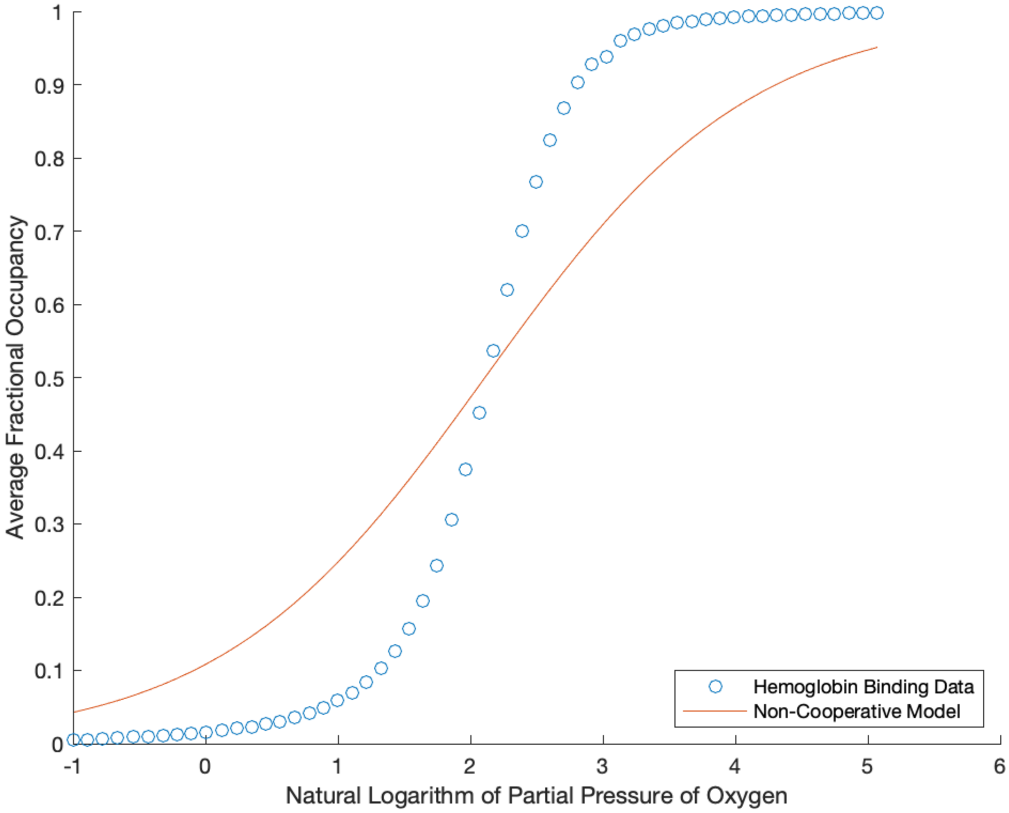
```
  0.1212
```

Using this `b`, we can use the above equation involving `x` and `<N>` to get the fit for the non-cooperative model.

In [28]:

```
x = ppo2.*b;
numerator = 4.*x+12.*x.^2+12.*x.^3+4.*x.^4;
denominator = 1+4.*x+6.*x.^2+4.*x.^3+x.^4;
N = numerator ./ denominator ./ 4;
scatter(ln_ppo2, frac_occupancy)
hold on
plot(ln_ppo2, N)
hold off
legend({"Hemoglobin Binding Data", "Non-Cooperative Model"}, "location", "sc
xlabel("Natural Logarithm of Partial Pressure of Oxygen")
ylabel("Average Fractional Occupancy")
```

Out[28]:



This fit doesn't appear right compared to the data.

The Cooperative Model (Pauling)

Now let's check the cooperative model. This specific model is called the Pauling Model. In this case we must fit both parameters, and we cannot assume `j=1`.

$$\langle N \rangle = \frac{4x + 12x^2 j + 12x^3 j^3 + 4x^4 j^6}{1 + 4x + 6x^2 j + 4x^3 j^3 + x^4 j^6} \tag{10}$$

Since we are fitting two parameters, we must work a little differently. Let's check many values of `b`, and cross check them with many values of `j`.

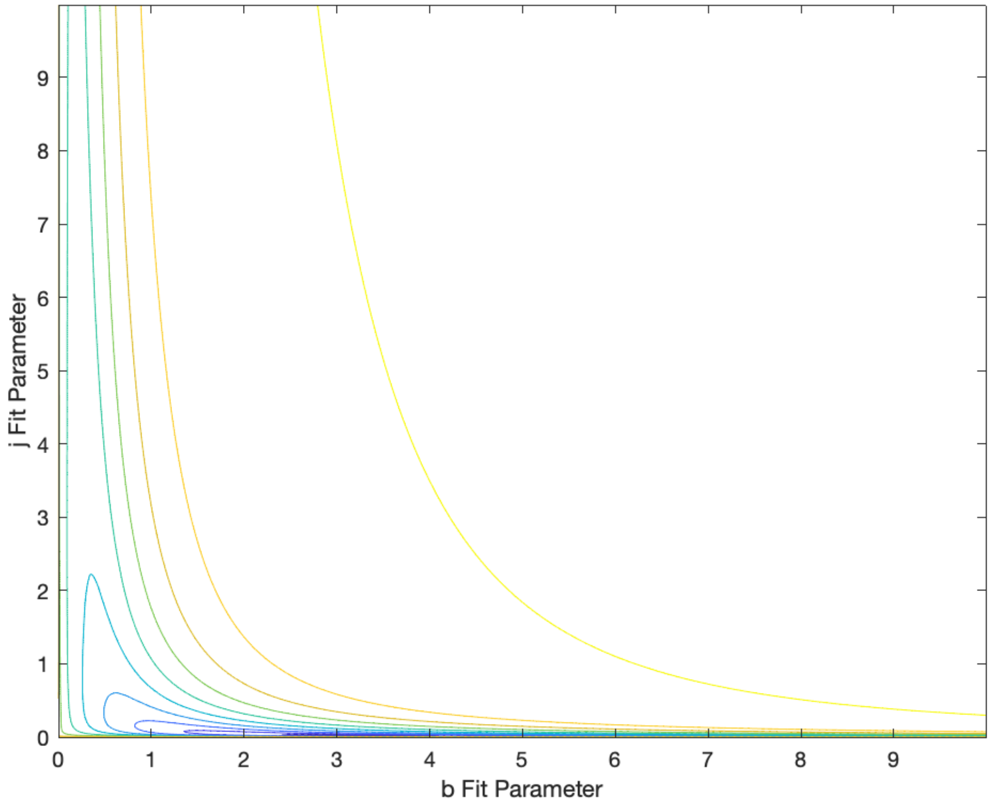
In [29]:

```
b = 0:.01:9.99;
j = 0:.01:9.99;
ppo2 = exp(ln_ppo2);
x = ppo2.*b;
N_hat_coop = [];
for k = 1:length(j)
    i = j(k);
    numerator = 4.*x+12.*(x.^2).*i+12.*(x.^3).*(i^3)+4.*(x.^4).*(i^6);
    denominator = 1+4.*x+6.*(x.^2).*i+4.*(x.^3).*(i^3)+x.^4.*(i^6);
    N_hat_coop(:,:,k) = numerator ./ denominator ./ 4;
end
```

In [30]:

```
error = sum(abs(frac_occupancy - N_hat_coop(:,:,:))) ./ length(ln_ppo2);
error_T = reshape(error, length(b), length(j));
contour(b, j, error_T)
xlabel('b Fit Parameter')
ylabel('j Fit Parameter')
```

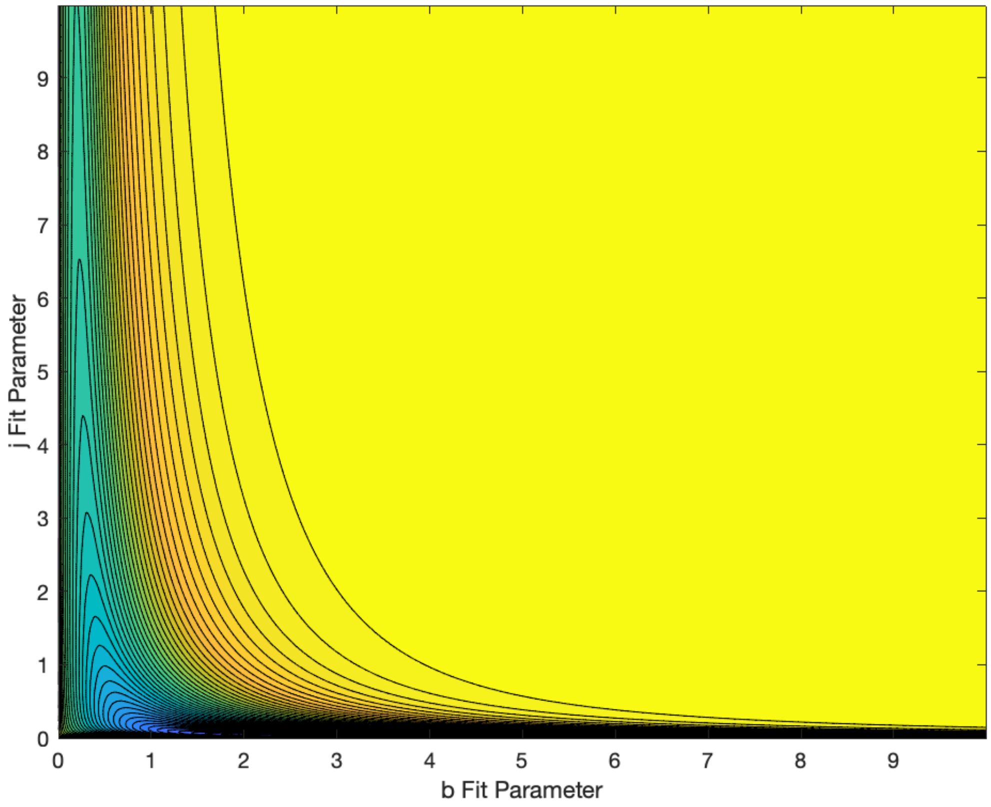
Out[30]:



In [31]:

```
error = sum(abs(frac_occupancy - N_hat_coop(:,:,:))) ./ length(ln_ppo2);
error_T = reshape(error, length(b), length(j));
contourf(b, j, error_T, 50)
xlabel('b Fit Parameter')
ylabel('j Fit Parameter')
```

Out[31]:



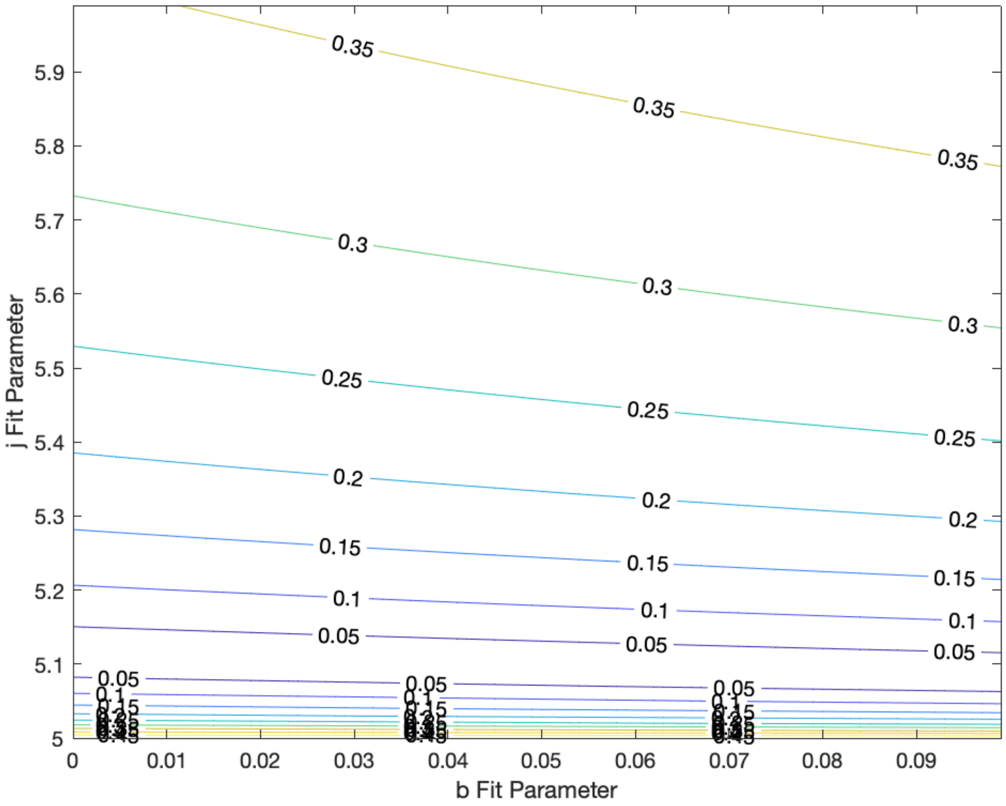
This contour plot shows us the mean absolute errors of different combinations of `b` and `j`. It seems that the error is minimized near the axes. It seems that the minimized error is when `j` is small, but that actually is not true. Upon closer inspection, the opposite is actually true, and the error is minimized when `b` is small (logical since most of the shape of the data comes from cooperation, which is `j`). If it is assumed that `j` is small, you actually find that the error is minimized when `j=1`, which is exactly the non-cooperative model. Although not shown in this notebook, the specific range we want to inspect is a `j` of [5,6) and a `b` of [0,0.1).

In [32]:

```
b = 0:.001:.099;
j = 5:.01:5.99;
ppo2 = exp(ln_ppo2);
x = ppo2.*b;
N_hat_coop = [];
for k = 1:length(j)
    i = j(k);
    numerator = 4.*x+12.*(x.^2).*i+12.*(x.^3).*(i^3)+4.*(x.^4).*(i^6);
    denominator = 1+4.*x+6.*(x.^2).*i+4.*(x.^3).*(i^3)+x.^4.*(i^6);
    N_hat_coop(:,:,k) = numerator ./ denominator ./ 4;
end
```

```
error = sum(abs(frac_occupancy - N_hat_coop(:,:,:))) ./ length(frac_occupanc
error_T = reshape(error, length(b), length(j));
contour(b, j, error_T, "ShowText", true)
xlabel('b Fit Parameter')
ylabel('j Fit Parameter')
```

Out[32]:

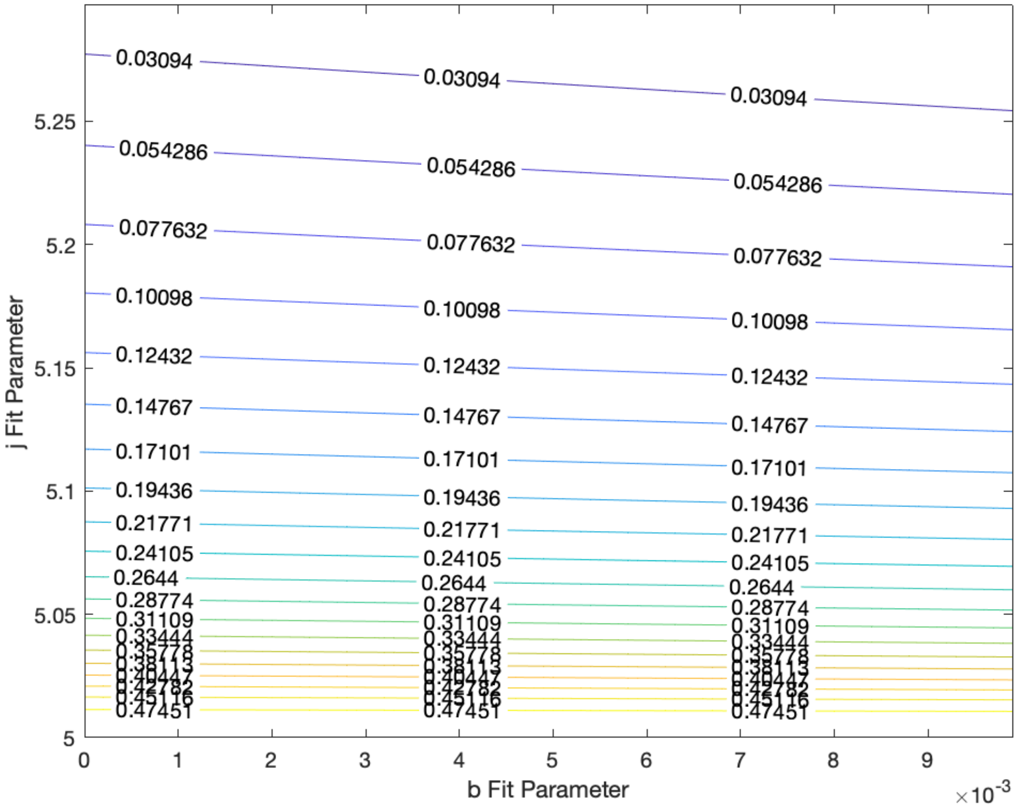


Let's inspect the range of the `0.05` error a little closer.

In [33]:

```
b = 0:.0001:.0099;
j = 5:.003:5.297;
ppo2 = exp(ln_ppo2);
x = ppo2.*b;
N_hat_coop = [];
for k = 1:length(j)
    i = j(k);
    numerator = 4.*x+12.*(x.^2).*i+12.*(x.^3).*(i^3)+4.*(x.^4).*(i^6);
    denominator = 1+4.*x+6.*(x.^2).*i+4.*(x.^3).*(i^3)+x.^4.*(i^6);
    N_hat_coop(:,:,k) = numerator ./ denominator ./ 4;
end
error = sum(abs(frac_occupancy - N_hat_coop(:,:,:))) ./ length(frac_occupanc
error_T = reshape(error, length(b), length(j));
contour(b, j, error_T, 20, "ShowText", true)
xlabel('b Fit Parameter')
ylabel('j Fit Parameter')
```

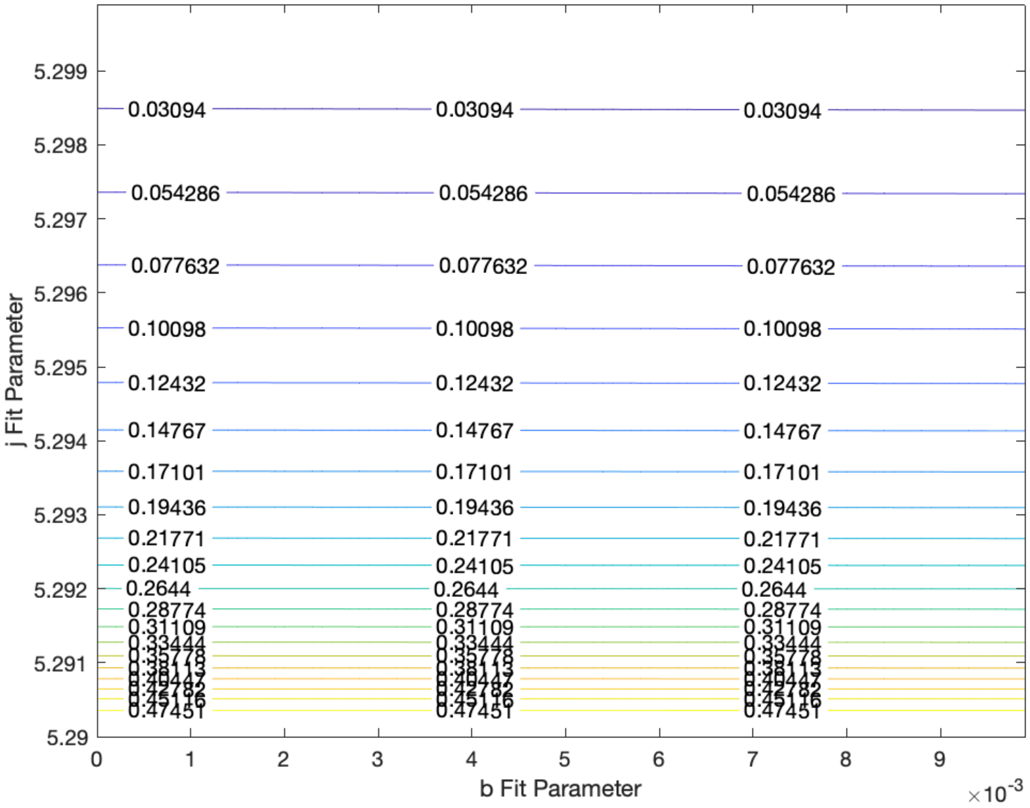
Out[33]:



In [34]:

```
b = 0:.0001:.0099;
j = 5.29:.0001:5.29995;
ppo2 = exp(ln_ppo2);
x = ppo2.*b;
N_hat_coop = [];
for k = 1:length(j)
    i = j(k);
    numerator = 4.*x+12.*(x.^2).*i+12.*(x.^3).*(i^3)+4.*(x.^4).*(i^6);
    denominator = 1+4.*x+6.*(x.^2).*i+4.*(x.^3).*(i^3)+x.^4.*(i^6);
    N_hat_coop(:,:,k) = numerator ./ denominator ./ 4;
end
error = sum(abs(frac_occupancy - N_hat_coop(:,:,:))) ./ length(frac_occupanc
error_T = reshape(error, length(b), length(j));
contour(b, j, error_T, 20, "ShowText", true)
xlabel('b Fit Parameter')
ylabel('j Fit Parameter')
```

Out[34]:



We can continue inspecting, but at this point let's choose a pair `(b, j)`. For simplicity, let's pick `b=0.01`, and then the corresponding `j=5.2985`.

In [35]:

```
j = 5.2985
b = .01
ppo2 = exp(ln_ppo2);
x = ppo2.*b;
numerator = 4.*x+12.*(x.^2).*j+12.*(x.^3).*(j.^3)+4.*(x.^4).*(j.^6);
denominator = 1+4.*x+6.*(x.^2).*j+4.*(x.^3).*(j.^3)+x.^4.*(j.^6);
N_coop = numerator ./ denominator ./ 4;
lowest_error = sum(abs(frac_occupancy - N_coop)) ./ length(ln_ppo2)
```

Out[35]:

```
j =
  5.2985
```

Out[35]:

```
b =
  0.0100
```

Out[35]:

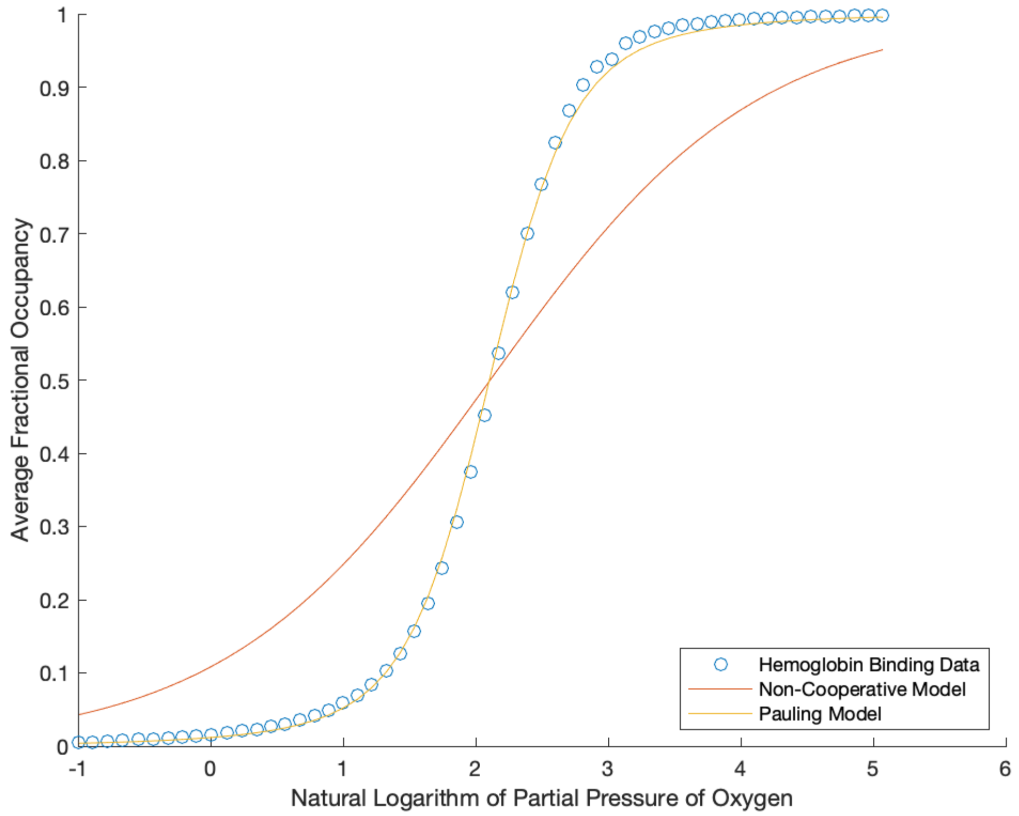
```
lowest_error =
  0.0077
```

In [36]:

```
scatter(ln_ppo2, frac_occupancy)
hold on
plot(ln_ppo2, N)
plot(ln_ppo2, N_coop)
hold off
xlabel("Natural Logarithm of Partial Pressure of Oxygen")
```

```
ylabel("Average Fractional Occupancy")
legend({"Hemoglobin Binding Data", "Non-Cooperative Model", "Pauling Model"}
xlabel("Natural Logarithm of Partial Pressure of Oxygen")
ylabel("Average Fractional Occupancy")
```

Out[36]:



This model seems to already be much better than the non-cooperative model.

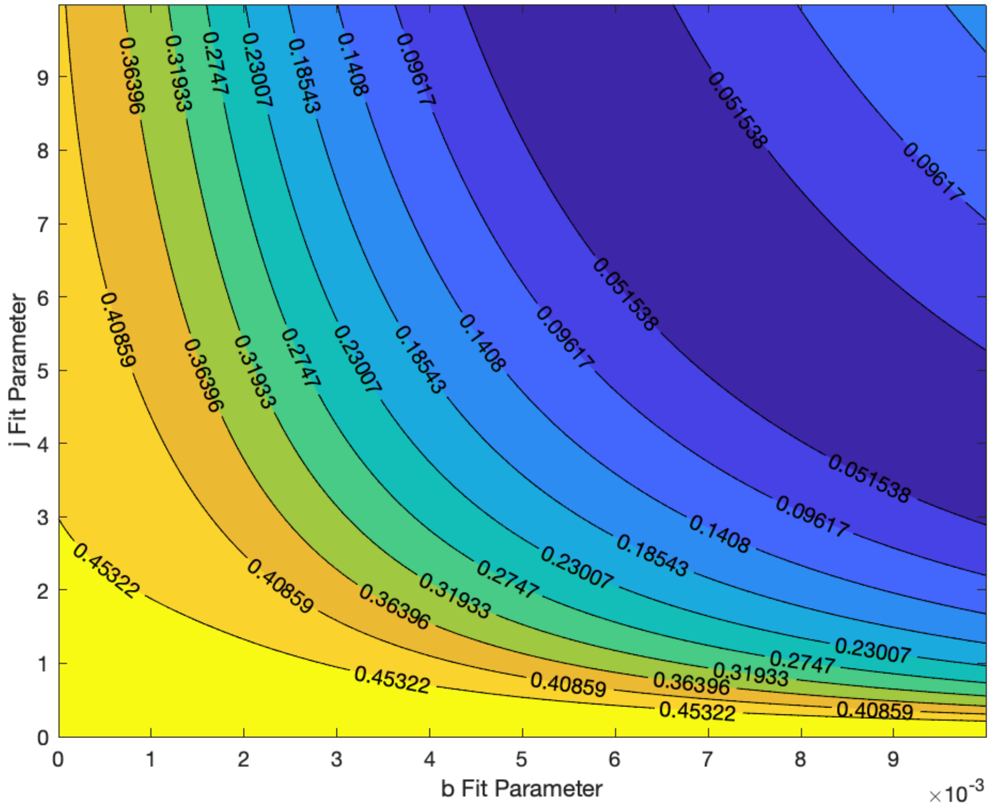
In [37]:

```
j = 0:.01:9.99;
b = 0:.00001:.00999;
ppo2 = exp(ln_ppo2);
x = ppo2.*b;
N_hat_coop = [];
for k = 1:length(j)
    i = j(k);
    numerator = 4.*x+12.*(x.^2).*i+12.*(x.^3).*(i^3)+4.*(x.^4).*(i^6);
    denominator = 1+4.*x+6.*(x.^2).*i+4.*(x.^3).*(i^3)+x.^4.*(i^6);
    N_hat_coop(:,:,k) = numerator ./ denominator ./ 4;
end
```

In [38]:

```
error = sum(abs(frac_occupancy - N_hat_coop(:,:,:))) ./ length(ln_ppo2);
error_T = reshape(error, length(b), length(j));
contourf(b, j, error_T, 10, "ShowText", "on")
xlabel('b Fit Parameter')
ylabel('j Fit Parameter')
```

Out[38]:



Above is a filled contour graph from that show the hidden band of low error that we don't see in the original contour plot. Finding this band was aided by the use of the `min` function.

The Cooperative Model (Adair)

The Adair model attempts to improve upon the Pauling model by not only taking into account interactions between pairs, but interactions between three and four binding sites. We will need to fit the following equation:

$$\langle N\rangle = \frac{4x + 12x^2j + 12x^3j^3k + 4x^4j^6k^4l}{1 + 4x + 6x^2j + 4x^3j^3k + x^4j^6k^4l} \tag{11}$$

However, fitting four parameters can be tedious, so we're going to run a Monte Carlo simulation instead. In essence, we're going to take a range of `b`, `j`, `k`, and `l` values, and then make a guess. If that guess has a lower error, then we use those values, and run that a large number of times.

In [39]:

```
b = 0:.01:9.99;
j = 0:.01:9.99;
k = 0:.01:9.99;
l = 0:.01:9.99;
best_b = 0;
best_j = 0;
```

```
best_k = 0;
best_l = 0;
lowest_error = 100; % large number
for a = 1:300000
    r = randi([1 1000], 1, 4);
    b0 = b(r(1));
    x = exp(ln_ppo2).*b0;
    j0 = j(r(2));
    k0 = k(r(3));
    l0 = l(r(4));
    numerator = (4*x)+(12*(x.^2).*j0)+(12*(x.^3).*(j0^3).*k0)+(4*(x.^4).*(j0
    denominator = 1+(4*x)+(6*(x.^2).*j0)+(4*(x.^3).*(j0^3).*k0)+((x.^4).*(j0
    N_hat = numerator ./ denominator ./ 4;
    error = sum(abs(frac_occupancy - N_hat)) ./ length(ln_ppo2);
    if error < lowest_error
        lowest_error = error;
        best_b = b0;
        best_j = j0;
        best_k = k0;
        best_l = l0;
    end
end
x = exp(ln_ppo2).*best_b;
j0 = best_j;
k0 = best_k;
l0 = best_l;
numerator = (4*x)+(12*(x.^2).*j0)+(12*(x.^3).*(j0^3).*k0)+(4*(x.^4).*(j0^6).
denominator = 1+(4*x)+(6*(x.^2).*j0)+(4*(x.^3).*(j0^3).*k0)+((x.^4).*(j0^6).
N_adair = numerator ./ denominator ./ 4;
```

In [40]:

```
best_b
best_j
best_k
best_l
```

Out[40]:

```
best_b =
  0.0200
```

Out[40]:

```
best_j =
  0.7000
```

Out[40]:

```
best_k =
  6.1600
```

Out[40]:

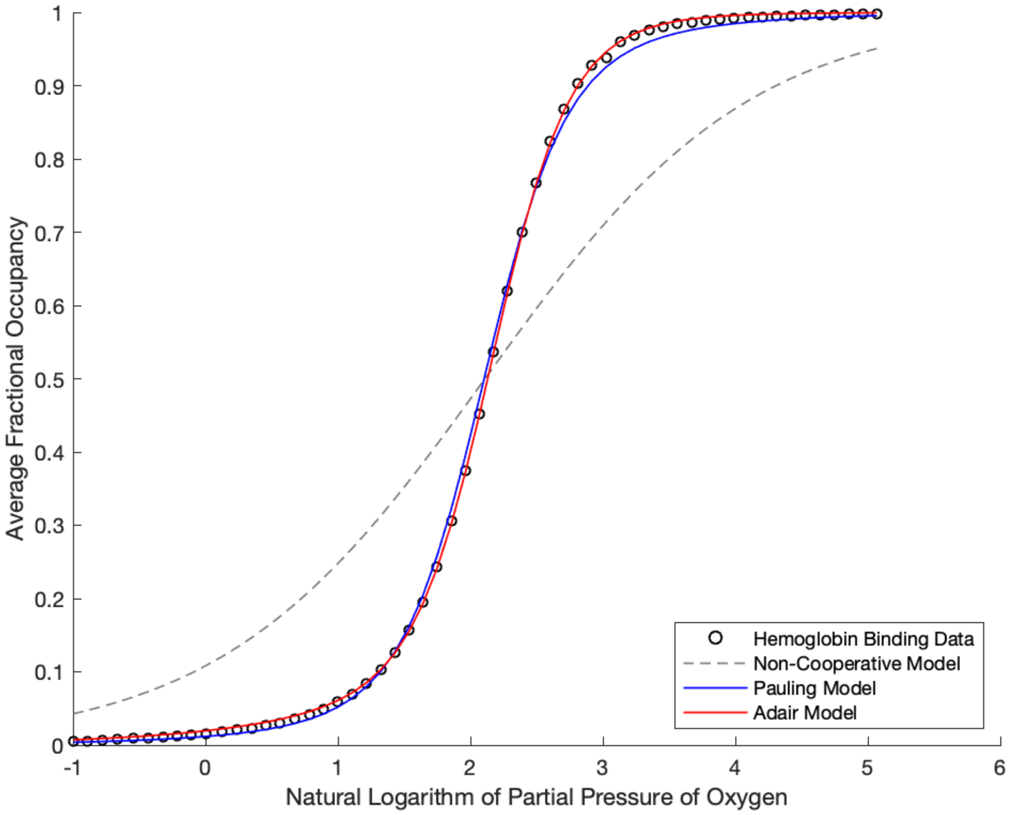
```
best_l =
  9.6600
```

In [41]:

```
scatter(ln_ppo2, frac_occupancy, 20, "o", "LineWidth", 1)
hold on
plot(ln_ppo2, N, "--", "LineWidth", .85)
plot(ln_ppo2, N_coop, "LineWidth", .85)
plot(ln_ppo2, N_adair, "LineWidth", .85)
hold off
xlabel("Natural Logarithm of Partial Pressure of Oxygen")
ylabel("Average Fractional Occupancy")
```

```
legend({"Hemoglobin Binding Data", "Non-Cooperative Model", "Pauling Model",
xlabel("Natural Logarithm of Partial Pressure of Oxygen")
ylabel("Average Fractional Occupancy")
mycolors = [0 0 0; .5 .5 .5; 0 0 1; 1 0 0];
ax = gca;
ax.ColorOrder = mycolors;
```

Out[41]:



In [42]:

```
error = sum(abs(frac_occupancy - N_adair)) ./ length(ln_ppo2)
```

Out[42]:

```
error =
  0.0031
```

We can see that this model improves on the Pauling model, but not super considerably.