In [155]:

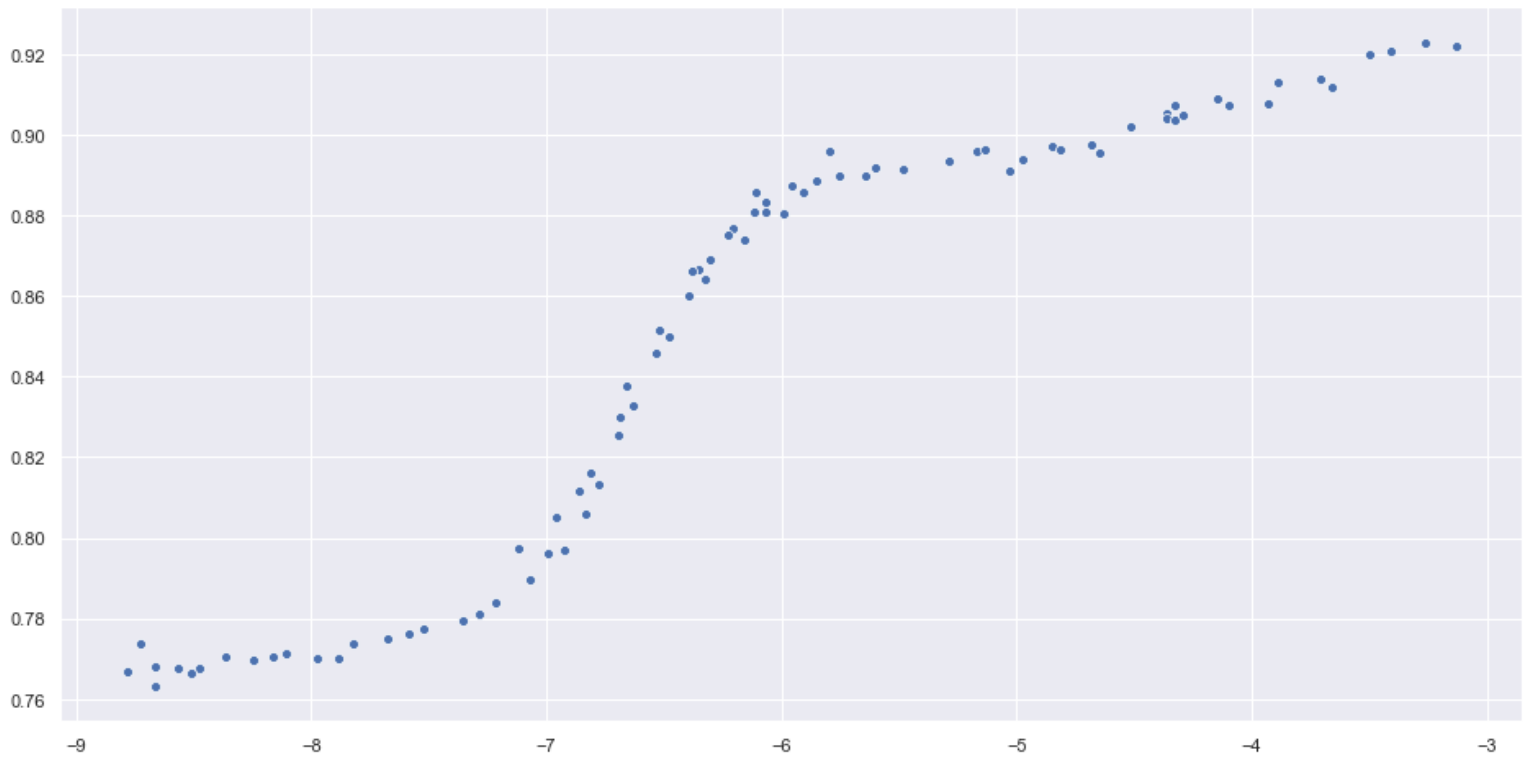
```
import seaborn as sns
import math
import numpy as np
import matplotlib.pyplot as plt
plt.rcParams['figure.figsize'] = (16, 8)
```

Ajustar polinomio de grado 10

Leer los datos y graficarlos

In [45]:

```
nist=[]
with open('nist.txt') as f:
    for line in f:
        nist.append( [float(i) for i in line.split()])
nist=np.array(nist)
ax=sns.scatterplot(x=nist[:,1],y=nist[:,0],)
```



Ajuste

Los coeficientes β son los siguientes:

In [159]:

```
poly,cov=np.polyfit(x=nist[:,1],y=nist[:,0],deg=10,cov=True)
X=np.linspace(-9,-3,200)
y_pred=np.polyval(x=X,p=poly)
y_fit=np.polyval(x=nist[:,1],p=poly)
print(poly[::-1])
```

```
[-1.4674895695328862e+03 -2.7721795072005034e+03 -2.3163710104027364
e+03
 -1.1279739060504180e+03 -3.5447822262226407e+02 -7.5124199364183198
e+01
 -1.0875317687044944e+01 -1.0622149513201322e+00 -6.7019113239696365
e-02
 -2.4678106994199226e-03 -4.0296251118195000e-05]
```

Con sus correspondientes errores estándar:

In [171]:

```
print(np.sqrt(np.diag(cov))[::-1])
```

```
[4.6785748495530662e+01 7.3201188914049993e+01 4.8805854143924662e+0
1
 1.7920160610158010e+01 3.8830193364130365e+00 5.0225876325099728e-0
1
 5.4895785419071585e-02 8.4079303106269036e-03 8.4281475970604818e-0
4
 4.3845508429966420e-05 9.3707893082698072e-07]
```

Resultado gráfico

In [132]:

```
ax=sns.scatterplot(x=nist[:,1],y=nist[:,0],alpha=.5)
ax=sns.lineplot(x=X,y=y_pred,ax=ax, color="coral")
```