REU 2022 Day 8: Challenge Answers

We are going to be comparing the Arctic temperature dataset we used this morning with Arctic sea ice area

- A. Load SIA time series data and compare time series of Arctic temperature and we used this morning
- B. Look at 5 coldest and warmest members in July during the period 1980-2000. Look at the same members with SIA in August, do you see any relationship?
- C. Look at Arctic temperatures in all 6 CMIP5 LEs and compute correlations with SIA in those models, also make scatter plots for each model, test out different time periods as well

A. Compare Arctic temperature time series for CanESM2 in July with sea ice area (SIA) in August

In [9]:
```
import xarray as xr
import numpy as np
import datetime
import matplotlib.pyplot as plt
import scipy.stats as stats
```

Load both sea ice area and Arctic temperature datasets for CanESM2

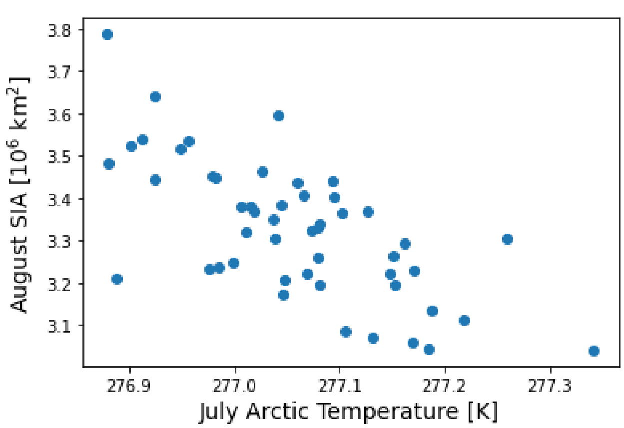
In [72]:
```
#Load SIA for all 6 models and make a variable for CanESM2 in August
CLIVAR_SIA = xr.open_dataset('CLIVAR_SIA_1850_2100_RCP85.nc')
CanESM2_SIA_Aug = CLIVAR_SIA['CanESM2'].sel(time=CLIVAR_SIA['time.month']==8).sel(member=slice(
1,50))
```

In [71]:
```
#Load Arctic temperatures for all 6 models and make a variable for CanESM2 in July
CLIVAR_Arctic_temp = xr.open_dataset('CLIVAR_Arctic_surface_temp_1850_2100_RCP85.nc')
CanESM2_Arctic_July = CLIVAR_Arctic_temp['CanESM2'].sel(time=CLIVAR_Arctic_temp['time.month']==
7).sel(member=slice(1,50))
```

Make a scatter plot of July Arctic temperature in 1980-2000 versus August SIA

In [73]:
```
July_temp = CanESM2_Arctic_July.sel(time=slice('1980','2000')).mean('time')
August_SIA = CanESM2_SIA_Aug.sel(time=slice('1980','2000')).mean('time')

plt.scatter(July_temp, August_SIA)
plt.ylabel(r'August SIA [10$^6$ km$^2$]', fontsize=14)
plt.xlabel('July Arctic Temperature [K]', fontsize=14);
```



Do you see a correlation? Does this make sense?

Yes, this makes sense. A higher July Arctic temperature generally shows a smaller sea ice area in August.

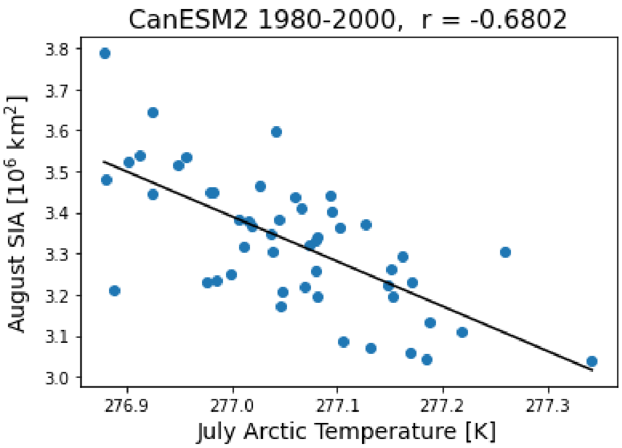
Compute the correlation, look back at Day 6 if you need to check how this is done. Add a line of best fit and the r value to the figure title

In [82]:
```
July_temp = CanESM2_Arctic_July.sel(time=slice('1980','2000')).mean('time')
August_SIA = CanESM2_SIA_Aug.sel(time=slice('1980','2000')).mean('time')
slope, intercept, r_value, p_value, std_err = stats.linregress(July_temp, August_SIA)

plt.scatter(July_temp, August_SIA)

x_vals = np.linspace(July_temp.min(), July_temp.max(), 2)
plt.plot(x_vals, x_vals*slope + intercept, c='k')
plt.title('CanESM2 1980-2000,  r = {}'.format(round(r_value,4)), fontsize=16)

plt.ylabel(r'August SIA [10$^6$ km$^2$]', fontsize=14)
plt.xlabel('July Arctic Temperature [K]', fontsize=14);
```



B. Now just look at the 5 highest and lowest temperature members, plot those against SIA for the same members

Find the member numbers of the highest and lowest Arctic temperatures in July 1980-2000. Hint look up the `.rank` function for xarray dataarrays

In [56]:
```
low_mem_list = np.array(CanESM2_Arctic_July['member'].where(CanESM2_Arctic_July.sel(time=slice(
'1980','2000')).mean('time').rank('member')<6, drop=True)).astype(int)
print(low_mem_list)
```

```
[ 5 27 38 40 47]
```

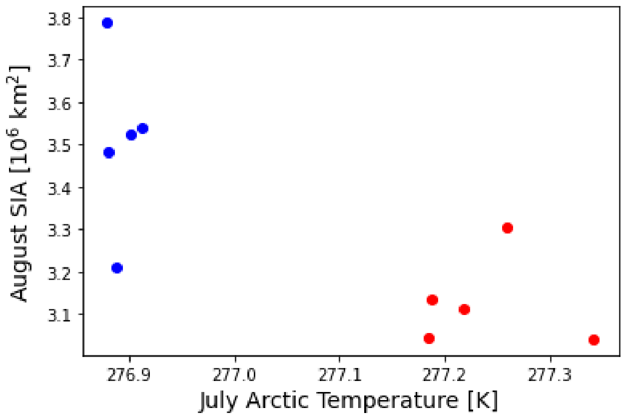
In [54]:
```
high_mem_list = np.array(CanESM2_Arctic_July['member'].where(CanESM2_Arctic_July.sel(time=slice
('1980','2000')).mean('time').rank('member')>45, drop=True)).astype(int)
print(high_mem_list)
```

```
[ 3  8 22 24 30]
```

Make a scatter plot to compare the July Arctic temperature and August SIA. Use 2 different colors for the high and low groups

In [59]:
```
plt.scatter(July_temp.sel(member=low_mem_list), August_SIA.sel(member=low_mem_list), c='b')
plt.scatter(July_temp.sel(member=high_mem_list), August_SIA.sel(member=high_mem_list), c='r')

plt.ylabel(r'August SIA [10$^6$ km$^2$]', fontsize=14)
plt.xlabel('July Arctic Temperature [K]', fontsize=14);
```



Do these look like 2 distinct groups in their SIA?

Not quite, there is some overlap between the blue and red dots on the y-axis

Calculate whether these two groups of SIA are statistically different at the 0.05 level, use `scipy.stats.ttest_ind`

In [61]:
```
stats.ttest_ind(August_SIA.sel(member=low_mem_list), August_SIA.sel(member=high_mem_list))
```

Out[61]:
```
Ttest_indResult(statistic=3.6881924918868467, pvalue=0.006146179348915911)
```

Yes, these groups are statistically different at the 0.05 level as the chance that they would randomly be this different is 0.0061 which is more unlikely than our threshold to reject the null hypothesis that the two groups are equal

C. Now compute correlations for the other 5 climate models in the datasets and plot scatter plots as you did for CanESM2 in part A

Make a variable for Arctic temperatures in July and SIA in August

In [127]:
```
#you can list the model names which are the xarray dataset variable names
model_names = np.array(list(CLIVAR_Arctic_temp.keys()))
print(model_names)
```

```
['CanESM2' 'CESM1' 'CSIRO_MK36' 'GFDL_CM3' 'GFDL_ESM2M' 'MPI_ESM1']
```

In [128]:
```
#each of these models have different numbers of ensemble members, in alphabetical order they ar
e as follows
mem_len = [50,40,30,20,30,100]
```

In [126]:
```
Arctic_temp_July = CLIVAR_Arctic_temp.sel(time=CLIVAR_Arctic_temp['time.month']==7)
SIA_August = CLIVAR_SIA.sel(time=CLIVAR_SIA['time.month']==8)
```

Repeat the same analysis as in A, make individual plots for each member to show a scatter plot with linear regression line and r value.
Make your code cell run for any of the model names and for any time period without rewriting the whole script

In [130]:

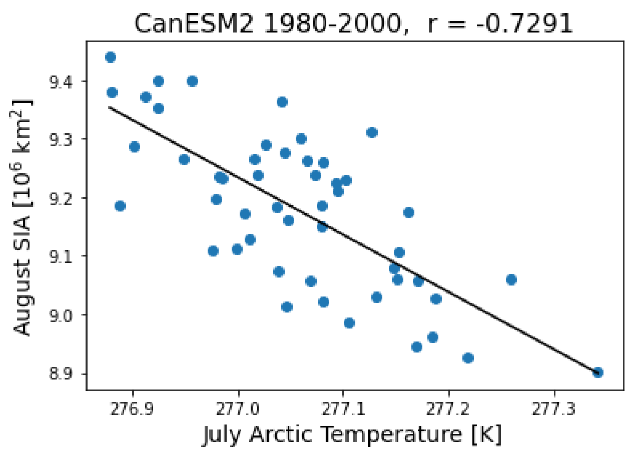
```
model_name = 'CanESM2'
start_yr = 1980
end_yr = 2000
#if we want to get rid of the annoying error of mean of an empty slice we need to only select t
he correct members for that model
max_mem = mem_len[np.where(model_names==model_name)[0][0]]

July_temp = Arctic_temp_July[model_name].sel(time=slice(str(start_yr),str(end_yr))).sel(member=
slice(1,max_mem)).mean('time')
August_SIA = CLIVAR_SIA[model_name].sel(time=slice(str(start_yr),str(end_yr))).sel(member=slice
(1,max_mem)).mean('time')
slope, intercept, r_value, p_value, std_err = stats.linregress(July_temp, August_SIA)

plt.scatter(July_temp, August_SIA)

x_vals = np.linspace(July_temp.min(), July_temp.max(), 2)
plt.plot(x_vals, x_vals*slope + intercept, c='k')
plt.title('{} {}-{},  r = {}'.format(model_name, start_yr, end_yr, round(r_value,4)), fontsize=
16)

plt.ylabel(r'August SIA [10$^6$ km$^2$]', fontsize=14)
plt.xlabel('July Arctic Temperature [K]', fontsize=14);
```



Extension, look at longer time periods, e.g. 1950-2100 - do you get better correlations?

In [125]:

```
r_vals_time_period = []
start_yr = 2000
end_yr = 2100

for model_name in model_names:

    max_mem = mem_len[np.where(model_names==model_name)[0][0]]

    July_temp = Arctic_temp_July[model_name].sel(time=slice(str(start_yr),str(end_yr))).mean('t
ime').sel(member=slice(1,max_mem))
    August_SIA = CLIVAR_SIA[model_name].sel(time=slice(str(start_yr),str(end_yr))).mean('time')
.sel(member=slice(1,max_mem))
    slope, intercept, r_value, p_value, std_err = stats.linregress(July_temp, August_SIA)
    r_vals_time_period.append(r_value)
print('{}-{} Model average r2 = {}'.format(start_yr, end_yr, np.mean(r_vals_time_period)**2))
```

```
2000-2100 Model average r2 = 0.7428024240916845
```