Problem Set 4: MAE 127 (Solutions)

For this problem set, you'll use data collected at three meteorological stations in San Diego County: San Miguel, Alpine, and Mt. Laguna. Download the data from the class website: http://www-pord.ucsd.edu/∼sgille/mae127/ps4.html or from the archived on-line data available on the UCSD server. The files contain daily minimum and maximum temperatures for the year 2004.

For all three problems, as always, please show your Matlab code.

1. Compute the correlation r for for all possible pairs of the three minimum temperature records and for all possible pairs of the three maximum temperature records. (The Matlab function "corrcoef" is an easy way to determine r, but check the help page to make sure you understand how to interpret its output.) Assuming that each observation is statistically independent, are these correlations statistically significant? Which pairs are most strongly correlated? Does this seem surprising?

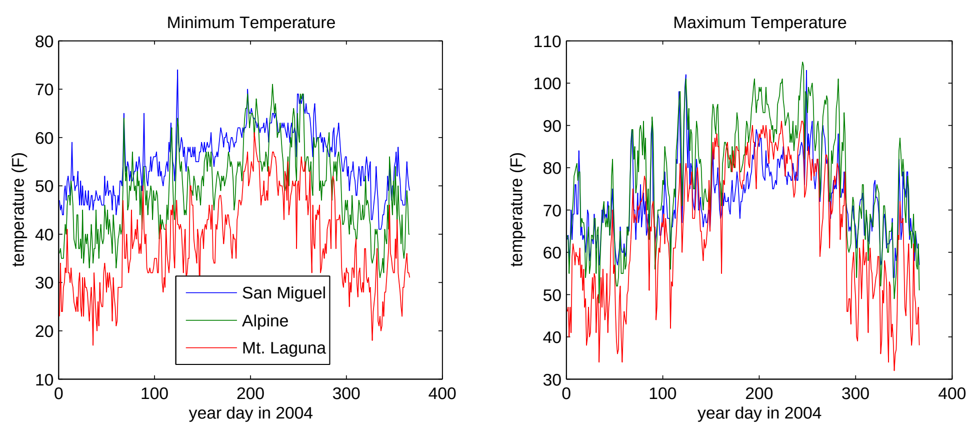

Figure 1: *Time series of (left) minimum daily temperatures and (right) maximum daily temperatures during 2004 from three recording stations in San Diego County.*

Figure 1 shows the time series from these three locations. Not surprisingly, all three undergo similar annual cycles. Here's code to compute the correlation coefficients:

```
load temperature.mat
c=corrcoef(alpine_min,sanmiguel_min); c_min(1)=c(1,2);
c=corrcoef(alpine_min,mtlaguna_min); c_min(2)=c(1,2);
c=corrcoef(sanmiguel_min,mtlaguna_min); c_min(3)=c(1,2);
c=corrcoef(alpine_max,sanmiguel_max); c_max(1)=c(1,2);
c=corrcoef(alpine_max,mtlaguna_max); c_max(2)=c(1,2);
c=corrcoef(sanmiguel_max,mtlaguna_max); c_max(3)=c(1,2);
sig_level=.95;
N=length(alpine_max);
significance= erfinv(sig_level)*sqrt(2/N)
```

The results show that:

Correlation Coefficients for Minima		
	San Miguel	Alpine
Alpine	0.89	
Mt. Laguna	0.84	0.86
Correlation Coefficients for Maxima		
	San Miguel	Alpine
Alpine	0.89	
Mt. Laguna	0.74	0.92

In this example, N is 366, and correlation coefficients have less than a 5% chance of being the result of random white noise when r exceeds $r_{sig} = 0.10$. *Thus if we assume that each daily measurement is statistically independent, then all six correlation coefficients are statistically significant.*

2. Compute and plot the autocovariance of the Alpine minimum temperature record. (The Matlab function "xcov" is good for this.) Where does the first zero crossing occur? If you interpret the first zero crossing as a decorrelation scale and use this to adjust the number of effective degrees of freedom N_{eff}, how does the 95% significance level r_{sig} change compared with your results in question 1?

In this data, the first zero crossing occurs only after many months, because temperature data vary slowly over the course of the year. This slow time-scale may not be relevant for understanding our problem, so you could also try determining N_{eff} by using twice the time interval required for the autocorrelation to drop from 1 to 0.5. How would this change the results?

Finally, speculate a little: What seems like a relevant time-scale to you?

The autocovariance can be computed in one line:

```
% compute autocovariance
autocov=xcov(alpine_min,'coeff');

% plot autocovariance
lag=-365:365;
plot(lag,autocov);
xlabel('time lag (days)'); ylabel('autocovariance');
title('Autocovariance for temperature minimum at Alpine');
```

Figure 2 shows the autocovariance for all lags and an enlargement for small lags. We can use the information form the autocovariance to help determine the number of degrees of freedom in our record.

```
% find zero crossings (from positive to negative)
lag(find(autocov(1:end-1)>0 & autocov(2:end)<0 & lag(1:end-1)'>0))
```

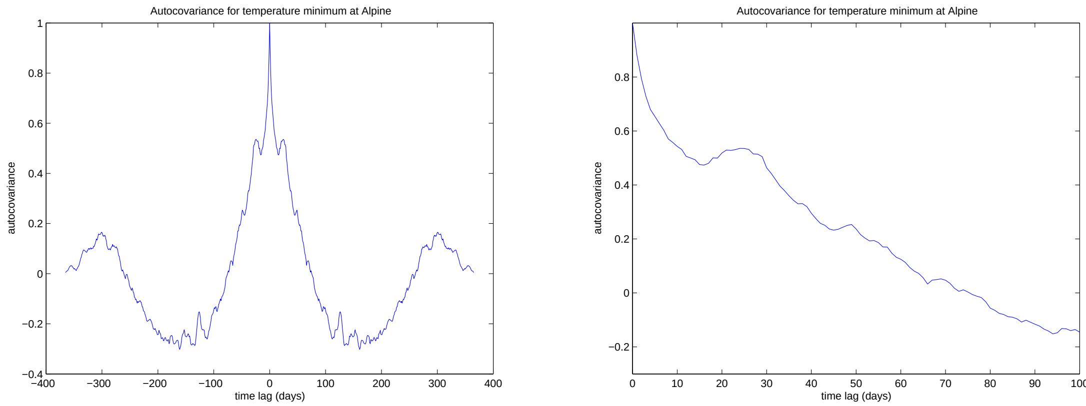

Figure 2: *Autocovariance of 2004 temperature minimum data from Alpine (left) for all lags, and (right) for small positive lags only.*

```
% find interval required for autocovariance to drop to 0.5
lag(find(autocov(1:end-1)>0.5 & autocov(2:end)<0.5 & lag(1:end-1)'>0))
```

The first zero crossing occurs after 75 days, so one might choose to set $N_{\rm eff} = N/75 = 366/75$. *In this case,* $r_{sig} = 0.89$, *which would imply that the autocovariances computed above are only marginally significant.*

The autocovariance drops more quickly to 0.5, after just 12 days, so one could also use $2 \times 12 = 24$ *days as a decorrelation scale, implying* $N_{\rm eff} = 366/24$. *This would imply* $r_{sig} = 0.50$.

You could also compute degrees of freedom by summing the autocovariance over all lags. (Formally, instead of computing:

$$\tau = n_d \delta_t = \sum_{n=-N}^{N} C_{\rm Alpine}(n)\delta t,$$

we should sum the product of two autocovariances:

$$\tau = n_d \delta_t = \sum_{n=-N}^{N} C_{\rm Mt.\ Laguna}(n) C_{\rm Alpine}(n)\delta t,$$

but the autocovariance from one location would give us a plausible guess.) This would imply a decorrelation scale around 40 to 50 days.

What's the right decorrelation scale? In this data, our calculation of the decorrelation scale is governed by the seasonal variability in the system. Since the seasonal cycle is big and repeatable, it makes the data look like it has a long and slow decorrelation scale. In fact, for statistical purposes we're probably more interested in a much shorter time scale associated with weather processes, perhaps around 3 or 4 days. To estimate this, we'd have to first remove the annual cycle from the data, which is what question 3 asks you to do.

3. Since the temperature data have a significant annual cycle, we might want to remove the annual cycle before doing further analysis. Using the Alpine minimum temperature data, least-squares fit a function of the form:

$$T = a + b\cos(2\pi t/366) + c\sin(2\pi t/366), \tag{1}$$

where T is temperature, and t is time in days.

The least-squares fit is done with the following Matlab commands:

```
A=[ones(size(year_day)) cos(2*pi*year_day/366) sin(2*pi*year_day/366)];
x=inv(A'*A)*A'*alpine_min

plot(year_day,[alpine_min A*x])
title('minimum temperatures at Alpine, 2004')
ylabel('temperature (F)')
xlabel('time (day of year)')
```

The results indicate that $a = 49.1066°F$, $b = -8.4013°F$, *and* $c = 3.6415°F$. *The value of a actually tells us that the mean minimum temperature at the Alpine meteorological station is 49.11°F, since cosine and sine have means of zero. The total amplitude of the annual cycle has an amplitude of* $\sqrt{b^2 + c^2} = 9.16°F$.