Homework 2 - Model answer:

1. **Diagnosing the model**
   Use the residuals and beta weights from the OLS estimation from homework 1. Write a function that provides some diagnostics:
   a. Calculate the root mean square error for each all voxel (p) across all time points (t).

   $$RMS_p = \sqrt{\sum_{t=1}^{T}\left(y_{t,p} - \hat{y}_{t,p}\right)^2 / T}$$

   Calculate and report the mean RMS for all gray and white matter voxels. Which one is higher and why?
   b. Plot the root-mean-square image. Where are there especially high residuals and why?
   c. Compute the root-mean-square-timeseries across all gray matter voxels (GM>0.5). Before computing the root-mean-square time series, it is recommended to standardize the residual-time-series of each voxel by dividing it by the root-mean-square-error for that voxel, so each voxel has the same weight in this computation.
   d. Using the residual time series, estimate and plot the auto-correlation function for temporal lags from t= 0 to 20, average for all gray-matter voxels and averaged for all white-matter voxels. What is the difference between gray and white matter? What could the reason for this difference be?
   e. To estimate the variance of your estimators, calculate the consistency of the regression weights for the fingers k=1…10, across *gray matter voxels* p, across all runs r=1…8, where $\bar{b}_{pk}$ is the mean regression coefficient for voxel p, finger k (across runs):

   $$R^2 = 1 - RSS / TSS$$

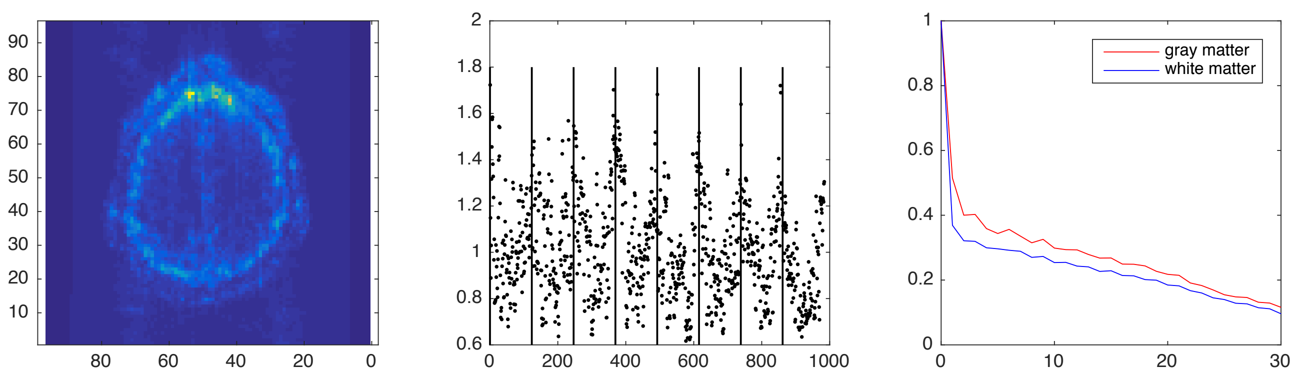
   $$TSS = \sum_p \sum_k \sum_r b_{pkr}^2$$

   $$RSS = \sum_p \sum_k \sum_r \left(b_{pkr} - \bar{b}_{pk}\right)^2$$

   The coefficient can be interpreted as the amount of variance in the in regression coefficients that is consistent across runs.

```
load dataset_1
load dataset_2

Y= reshape(Y,96*96,984)';

[T,P]=size(Y);

% Do a GLM analysis
for r=1:8
    X=[Xtask(:,:,r) Xintercept(:,:,r)];
    B(:,:,r)=pinv(X)*Y(run==r,:);
    R(run==r,:)=Y(run==r,:)-X*B(:,:,r);
end;
diagnose(R,B,GM,WM); % Diagnose residuals
```

```
function diagnose(R,B,GM,WM);
indx{1} = find(GM(:)>0.5);
indx{2} = find(WM(:)>0.5);
[T,P]=size(R);

% a/b. Plot ResMS Image
RMS = sqrt(sum(R.^2,1)/984);
subplot(1,3,1);
imagesc(reshape(RMS,96,96));
set(gca,'YDir','normal','XDir','reverse');
axis equal;
fprintf('Residual gray matter: %2.2f\n',mean(RMS(indx{1})));
fprintf('Residual white matter: %2.2f\n',mean(RMS(indx{2})));

% c. Plot ResMS time series
R=bsxfun(@rdivide,R,RMS);
RMST = sqrt(sum(R(:,indx{1}).^2,2))/length(indx{1});
subplot(1,3,2);
t=[1:984];
plot(t,RMST,'k.');
drawline([0.5:123:984]);

% d. Estimate acf on gray and white matter
subplot(1,3,3);
numlags=30;
for j=1:2
    XC=[];
    for i=1:length(indx{j})
        [XC(:,i),lags]=xcorr(R(:,indx{j}(i)),numlags);
    end;
    xs(:,j)=mean(XC(numlags+1:end,:),2)/T.^2;
end;
plot([0:numlags]',xs(:,1),'r',[0:numlags]',xs(:,2),'b');
legend({'gray matter','white matter'});

% f. Calculate pattern consistency
gB  = B(1:10,indx{1},:); % Pick out the gray matter Beta for the
10 fingers
TSS = sum(sum(sum(gB.^2)));
mB = mean(gB,3);
res = bsxfun(@minus,gB,mB);
RSS = sum(sum(sum(res.^2)));
R2 = 1-RSS/TSS;
fprintf('Consistency of Betas %2.2f\n',R2);
```



a. The RMS for gray and white matter are:
Residual gray matter: 17.77
Residual white matter: 11.07
The RMS is higher in gray matter, as the signal is higher here as well. This is partly due to signal-dependent noise, partly due to the fact that there are more vascular artifacts in gray matter.

b. The RMS image (left panel) shows higher residuals in gray than in white matter, and especially high residuals in areas where there the raw signal was high.

c. The residual time series (middle panel) show very few outlying observations. However, the RMS is high for the beginning and end of each run. This is due to the uncorrected time series.

d. The autocorrelation function (right panel) shows strong dependence of residuals across many observations. The acf is slightly higher in gray matter than in white matter - indicating that the low-frequency drifts are larger for gray matter.

f. The consistency of the activation patterns in gray matter are 0.21. That means that only 21% of the variance is systematic across runs - the rest is noise.

2. **Improve the model by temporal filtering.**
   a. Remove the low frequency trends from the data by first regressing the data against the high-pass filter regressors (hpf). You can get the residuals by first calculating the residual-forming matrix for the high-pass filter regressors H = Xhpf:
   $\mathbf{R} = \mathbf{I} - \mathbf{H}\left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathbf{H}^T$ a

   , where **I** an Identity matrix of size 984x984.
   The filtered data then is simply:
   $\tilde{\mathbf{Y}} = \mathbf{RY}$.
   Plot the same voxels as in homework 1. What do you notice?

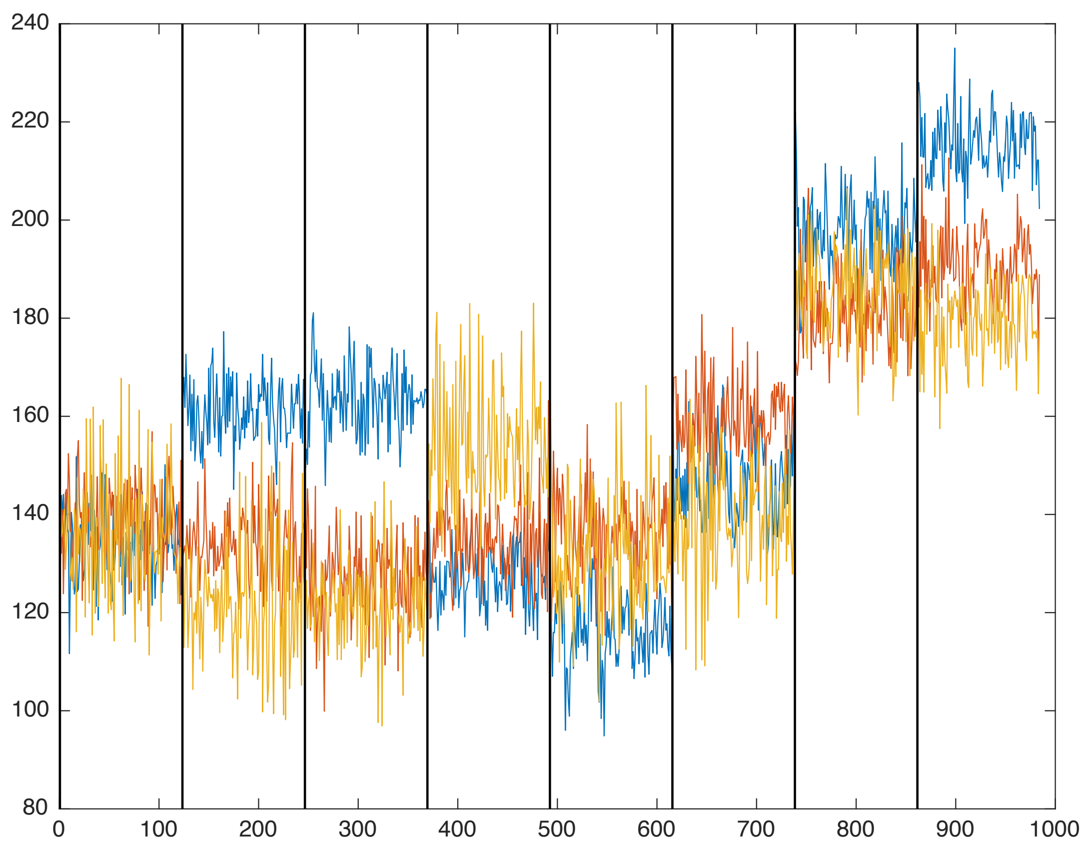
   b. Remove the low frequency trends from your task-related regressors (Xtask) by premultiplying R and plot the filtered and unfiltered first regressor from the first run. How is the regressor changed through the filtering process?

   c. Re-estimate the regression coefficients using filtered data and filtered design using OLS. Use your function from (1) to diagnose the residuals. What do you notice? Are the residuals and the consistency of the estimators improved?

```
% a. Filter the data
for r=1:8
    H = Xhpf(:,:,r);        % Low frequency regressors
    Tb=123;                 % Length of one run
    Re = (eye(Tb)-H*inv(H'*H)*H');  % residual forming matrix
    Yt(run==r,:) = Re*Y(run==r,:);                  % Filter data
end;
figure(1);
plot(Yt(:,[1869 1958 2056]));       % Plot some voxels
```

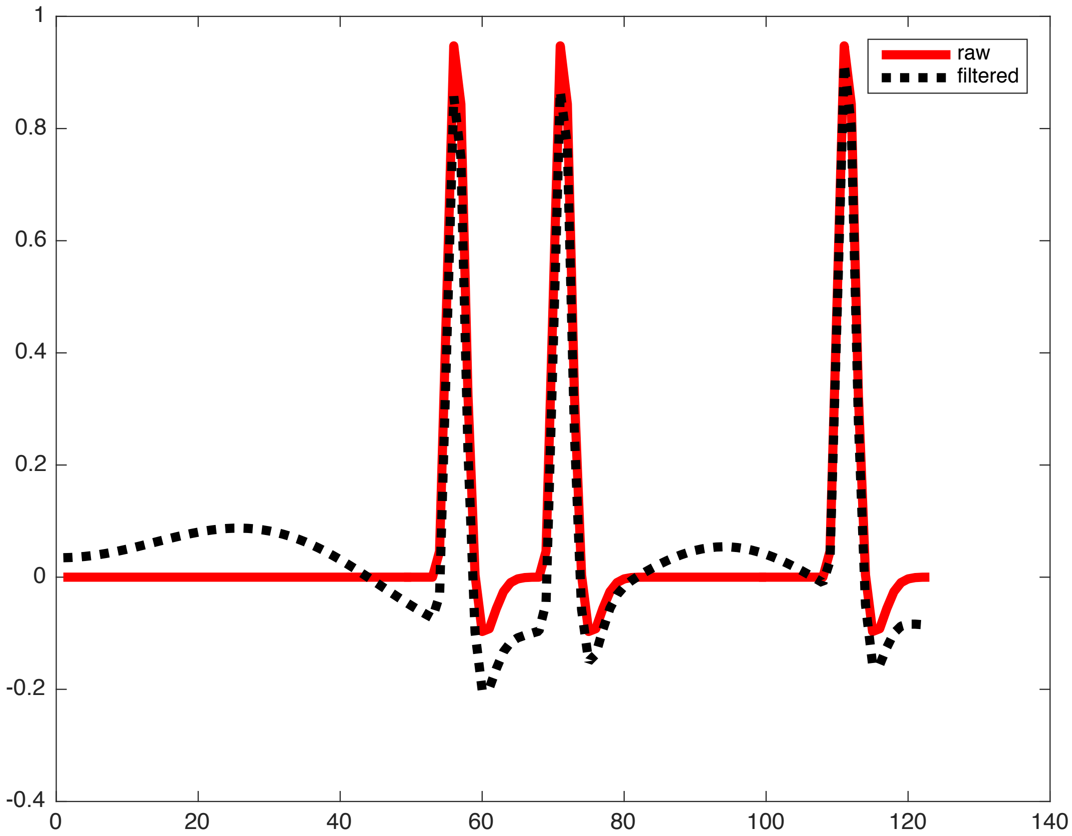
```
drawline([0.5:123:984]); % Draw lines at boundaries of runs (note:
you may not have that function)

% b. Filter and plot first regressor
figure(2);
X=[Xtask(:,:,1) Xintercept(:,:,1)];  % Build the design matrix
Xt = Re*X;                           % Filter the regressors
t=[1:Tb]';
h=plot(t,X(:,1),'r',t,Xt(:,1),'k:');      % Plot the first
regressor
set(h,'LineWidth',4);
legend({'raw','filtered'});

% c. Do the GLM and diagnose
for r=1:8
    X=[Xtask(:,:,r) Xintercept(:,:,r)];
    Xt = Re*X;
    B(:,:,r)=inv(Xt'*Xt)*Xt'*Yt(run==r,:);
    R(run==r,:)=Yt(run==r,:)-Xt*B(:,:,r);
end;
figure(3);
diagnose(R,B,GM,WM);
```

a.



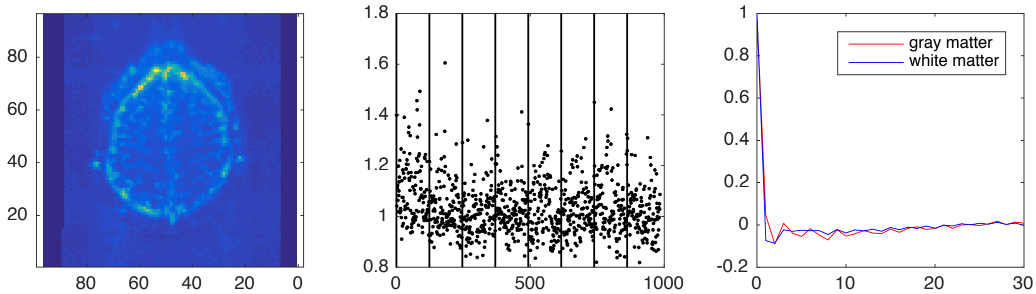
The Figure shows the time series of the same voxels as plotted in homework 1, only this time filtered. As we can see, the drifts within runs are removed. There exist only *the mean differences between runs, which will be accounted for by the different intercepts between runs.*

b.



The unfiltered regressor (red) models the three occurrences of the trial in the run. The filtered regressor (dashed line) changes the regressor, so that it mostly contrasts each activity to the local neighbourhood - for example the regressor is especially negative between the two tightly spaced events. However, the particular filter used induces some weird wiggles in the beginning of the run.

c.

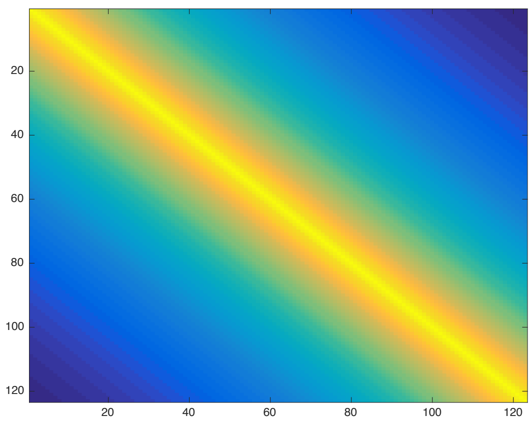


The diagnostics look much better for than for the OLS (homework 1).
The residuals have reduced.
Residual gray matter: 10.83
Residual white matter: 7.66

The autocorrelation of the residual has basically disappeared (right panel)
The consistency of the betas has increased to 0.42

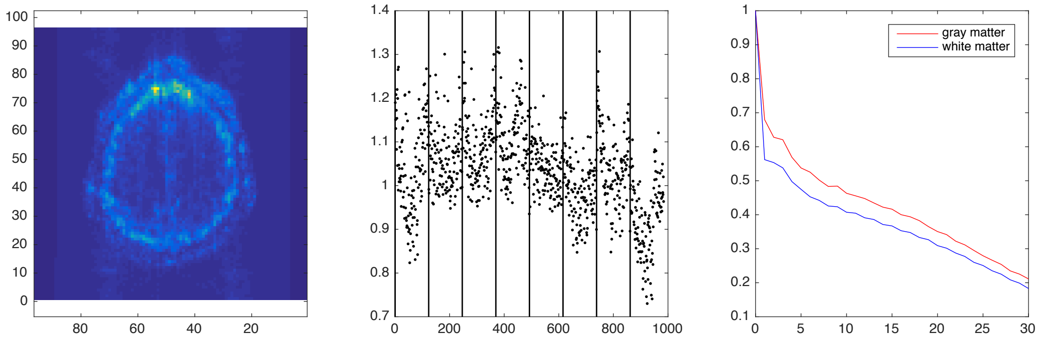
3. **General-least-squares estimation**
   Given the residuals, a good choice for the TxT variance-covariance matrix Φ for each run is a Toeplitz matrix, where the value of the nth diagonal is given by:
   $\phi_n = 0.7 * 0.98^n + 0.3$ and $\phi_0 = 1$.
   Generate and plot this matrix. Then use it in a General-least-square (GLS) estimation.
   $\hat{\mathbf{b}} = \left(\mathbf{X}^T \Phi^{-1} \mathbf{X}\right)^{-1} \mathbf{X}^T \Phi^{-1} \mathbf{Y}$
   Diagnose and compare the model quality using your diagnostic function from 1.

```
% Make the autocorrelation matrix
n=[0:122];
r = 0.7*(0.98.^n)+0.3;
r(1)=1;
Phi=toeplitz(r);   % Generate toeplitz matrix
figure(4);
imagesc(Phi);
iPhi=inv(Phi);
% Do the GLS and diagnose
for r=1:8
    X=[Xtask(:,:,r) Xintercept(:,:,r)];
    B(:,:,r)=inv(X'*iPhi*X)*X'*iPhi*Y(run==r,:);
    R(run==r,:)=Y(run==r,:)-X*B(:,:,r);
end;
figure(5);
diagnose(R,B,GM,WM);
```



The variance-covariance matrix is a banded (Toeplitz) matrix with large values close to the diagonal, which then fall off.

gray: 21.15



White: 13.00
Consistency of Betas 0.4565

The raw residuals are higher than what we got from the the OLS - however this is normal as they are unfiltered. How would we get filtered residuals? However, the consistency of the regression estimates has gotten better, indicating that this is a better filter than in 2.

4. The residual forming matrix for a regression is:

$$\mathbf{R} = \mathbf{I} - \mathbf{X}\left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T$$

Because $\mathbf{y} = \mathbf{Xb} + \varepsilon$, the residuals of Y are

$$r = \mathbf{Ry} = \mathbf{RXb} + \mathbf{R}\varepsilon$$

The first term is zero. Thus the variance-covariance matrix of the residuals is

$$\mathrm{var}(\mathbf{r}) = E\left(\mathbf{rr}^T\right) = E\left(\mathbf{R}\varepsilon\varepsilon^T\mathbf{R}^T\right) = \mathbf{R}\,\mathrm{var}(\varepsilon)\mathbf{R}^T$$