Train deepGLM for DirectMarketing Data

Load DirectMarketing dataset. The data has following information:

- Training data: 900 observations
- Testing data: 100 observations
- Number of covariates: 10

```
clear
clc
load('../Data/DirectMarketing.mat')
```

Training Phase

Now start training deepGLM with some settings. By default, if **'distribution'** option is not specified then deepGLMfit will treat y as continuous responses

```
nn = [5,5];
lb = true;
mdl = deepGLMfit(X,y,...
                 'Network',nn,...
                 'Lrate',0.01,...
                 'Verbose',10,...             % Display training result each iteration
                 'BatchSize',size(X,1),...   % Use entire training data as mini-batch
                 'MaxEpoch',10000,...
                 'Patience',50,...           % Higher patience values could lead to overfitti
                 'Lowerbound',lb,...         % Use lowerbound as early stopping condition and
                 'Seed',100);
```

```
Initial LB: -2.835
---------- Training Phase ----------
Epoch: 9   -   Current LB: -2.2324
Epoch: 19   -   Current LB: -2.1807
Epoch: 29   -   Current LB: -2.1808
Epoch: 39   -   Current LB: -2.1699
Epoch: 49   -   Current LB: -2.1604
Epoch: 59   -   Current LB: -2.1508
Epoch: 69   -   Current LB: -2.1404
Epoch: 79   -   Current LB: -2.1293
Epoch: 89   -   Current LB: -2.118
Epoch: 99   -   Current LB: -2.1061
Epoch: 109   -   Current LB: -2.0927
Epoch: 119   -   Current LB: -2.0782
Epoch: 129   -   Current LB: -2.0636
Epoch: 139   -   Current LB: -2.0476
Epoch: 149   -   Current LB: -2.0293
Epoch: 159   -   Current LB: -2.0093
Epoch: 169   -   Current LB: -1.9873
Epoch: 179   -   Current LB: -1.9627
Epoch: 189   -   Current LB: -1.9372
Epoch: 199   -   Current LB: -1.9122
Epoch: 209   -   Current LB: -1.8878
Epoch: 219   -   Current LB: -1.8658
Epoch: 229   -   Current LB: -1.8435
```

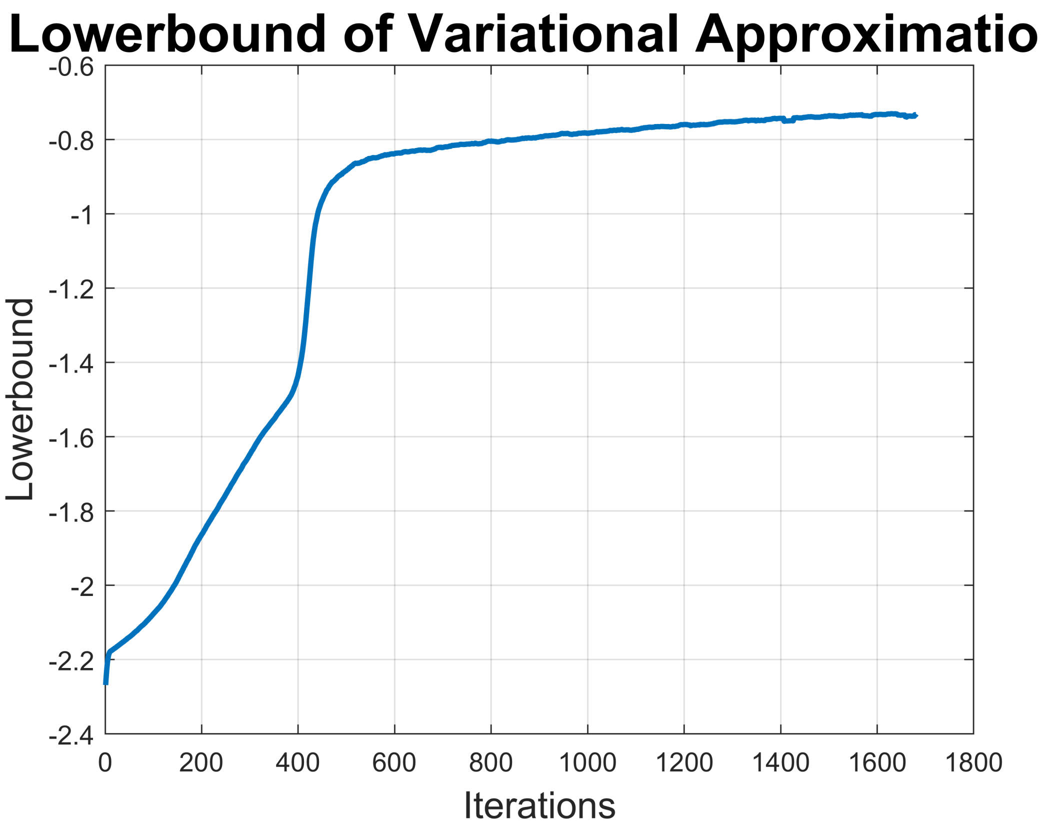
```
Epoch: 239   -   Current LB: -1.8222
Epoch: 249   -   Current LB: -1.8012
Epoch: 259   -   Current LB: -1.779
Epoch: 269   -   Current LB: -1.7578
Epoch: 279   -   Current LB: -1.7356
Epoch: 289   -   Current LB: -1.7136
Epoch: 299   -   Current LB: -1.6917
Epoch: 309   -   Current LB: -1.6698
Epoch: 319   -   Current LB: -1.6494
Epoch: 329   -   Current LB: -1.6282
Epoch: 339   -   Current LB: -1.6064
Epoch: 349   -   Current LB: -1.5872
Epoch: 359   -   Current LB: -1.5711
Epoch: 369   -   Current LB: -1.5545
Epoch: 379   -   Current LB: -1.5362
Epoch: 389   -   Current LB: -1.519
Epoch: 399   -   Current LB: -1.5013
Epoch: 409   -   Current LB: -1.4779
Epoch: 419   -   Current LB: -1.4391
Epoch: 429   -   Current LB: -1.369
Epoch: 439   -   Current LB: -1.2418
Epoch: 449   -   Current LB: -1.0966
Epoch: 459   -   Current LB: -1.0092
Epoch: 469   -   Current LB: -0.96584
Epoch: 479   -   Current LB: -0.93661
Epoch: 489   -   Current LB: -0.91779
Epoch: 499   -   Current LB: -0.90572
Epoch: 509   -   Current LB: -0.89397
Epoch: 519   -   Current LB: -0.88458
Epoch: 529   -   Current LB: -0.87365
Epoch: 539   -   Current LB: -0.86395
Epoch: 549   -   Current LB: -0.86242
Epoch: 559   -   Current LB: -0.85754
Epoch: 569   -   Current LB: -0.85198
Epoch: 579   -   Current LB: -0.84926
Epoch: 589   -   Current LB: -0.84606
Epoch: 599   -   Current LB: -0.84195
Epoch: 609   -   Current LB: -0.84048
Epoch: 619   -   Current LB: -0.83881
Epoch: 629   -   Current LB: -0.83632
Epoch: 639   -   Current LB: -0.83385
Epoch: 649   -   Current LB: -0.83352
Epoch: 659   -   Current LB: -0.83125
Epoch: 669   -   Current LB: -0.82912
Epoch: 679   -   Current LB: -0.82882
Epoch: 689   -   Current LB: -0.82921
Epoch: 699   -   Current LB: -0.82753
Epoch: 709   -   Current LB: -0.82201
Epoch: 719   -   Current LB: -0.82182
Epoch: 729   -   Current LB: -0.82001
Epoch: 739   -   Current LB: -0.81632
Epoch: 749   -   Current LB: -0.81527
Epoch: 759   -   Current LB: -0.81339
Epoch: 769   -   Current LB: -0.81271
Epoch: 779   -   Current LB: -0.81159
Epoch: 789   -   Current LB: -0.81104
Epoch: 799   -   Current LB: -0.81118
Epoch: 809   -   Current LB: -0.80694
Epoch: 819   -   Current LB: -0.80517
Epoch: 829   -   Current LB: -0.806
Epoch: 839   -   Current LB: -0.80523
Epoch: 849   -   Current LB: -0.80316
Epoch: 859   -   Current LB: -0.8015
Epoch: 869   -   Current LB: -0.8009
Epoch: 879   -   Current LB: -0.799
```

```
Epoch: 889   -   Current LB: -0.79714
Epoch: 899   -   Current LB: -0.79607
Epoch: 909   -   Current LB: -0.79592
Epoch: 919   -   Current LB: -0.79376
Epoch: 929   -   Current LB: -0.79128
Epoch: 939   -   Current LB: -0.79025
Epoch: 949   -   Current LB: -0.78869
Epoch: 959   -   Current LB: -0.78706
Epoch: 969   -   Current LB: -0.78443
Epoch: 979   -   Current LB: -0.78395
Epoch: 989   -   Current LB: -0.78618
Epoch: 999   -   Current LB: -0.78427
Epoch: 1009   -   Current LB: -0.78283
Epoch: 1019   -   Current LB: -0.78379
Epoch: 1029   -   Current LB: -0.78216
Epoch: 1039   -   Current LB: -0.77935
Epoch: 1049   -   Current LB: -0.77923
Epoch: 1059   -   Current LB: -0.77831
Epoch: 1069   -   Current LB: -0.77609
Epoch: 1079   -   Current LB: -0.77577
Epoch: 1089   -   Current LB: -0.7738
Epoch: 1099   -   Current LB: -0.77438
Epoch: 1109   -   Current LB: -0.77522
Epoch: 1119   -   Current LB: -0.77375
Epoch: 1129   -   Current LB: -0.77121
Epoch: 1139   -   Current LB: -0.76837
Epoch: 1149   -   Current LB: -0.76717
Epoch: 1159   -   Current LB: -0.76615
Epoch: 1169   -   Current LB: -0.7658
Epoch: 1179   -   Current LB: -0.76544
Epoch: 1189   -   Current LB: -0.76609
Epoch: 1199   -   Current LB: -0.76552
Epoch: 1209   -   Current LB: -0.76228
Epoch: 1219   -   Current LB: -0.76006
Epoch: 1229   -   Current LB: -0.76107
Epoch: 1239   -   Current LB: -0.76143
Epoch: 1249   -   Current LB: -0.76051
Epoch: 1259   -   Current LB: -0.76007
Epoch: 1269   -   Current LB: -0.76041
Epoch: 1279   -   Current LB: -0.75797
Epoch: 1289   -   Current LB: -0.75445
Epoch: 1299   -   Current LB: -0.75347
Epoch: 1309   -   Current LB: -0.75259
Epoch: 1319   -   Current LB: -0.75294
Epoch: 1329   -   Current LB: -0.75236
Epoch: 1339   -   Current LB: -0.75043
Epoch: 1349   -   Current LB: -0.74836
Epoch: 1359   -   Current LB: -0.74835
Epoch: 1369   -   Current LB: -0.75
Epoch: 1379   -   Current LB: -0.74852
Epoch: 1389   -   Current LB: -0.74693
Epoch: 1399   -   Current LB: -0.74573
Epoch: 1409   -   Current LB: -0.74371
Epoch: 1419   -   Current LB: -0.74341
Epoch: 1429   -   Current LB: -0.7507
Epoch: 1439   -   Current LB: -0.74997
Epoch: 1449   -   Current LB: -0.74216
Epoch: 1459   -   Current LB: -0.74218
Epoch: 1469   -   Current LB: -0.74047
Epoch: 1479   -   Current LB: -0.73909
Epoch: 1489   -   Current LB: -0.74009
Epoch: 1499   -   Current LB: -0.74046
Epoch: 1509   -   Current LB: -0.73851
Epoch: 1519   -   Current LB: -0.73667
Epoch: 1529   -   Current LB: -0.73684
```

```
Epoch: 1539   -   Current LB: -0.73753
Epoch: 1549   -   Current LB: -0.73744
Epoch: 1559   -   Current LB: -0.73644
Epoch: 1569   -   Current LB: -0.73514
Epoch: 1579   -   Current LB: -0.73467
Epoch: 1589   -   Current LB: -0.73477
Epoch: 1599   -   Current LB: -0.73627
Epoch: 1609   -   Current LB: -0.73596
Epoch: 1619   -   Current LB: -0.73294
Epoch: 1629   -   Current LB: -0.73281
Epoch: 1639   -   Current LB: -0.73269
Epoch: 1649   -   Current LB: -0.73089
Epoch: 1659   -   Current LB: -0.73135
Epoch: 1669   -   Current LB: -0.73494
Epoch: 1679   -   Current LB: -0.73811
Epoch: 1689   -   Current LB: -0.73717
Epoch: 1699   -   Current LB: -0.73359
---------- Training Completed! ----------
Number of iteration:1702
LBBar best: -0.7307
Training time: 14.2739s
```

Now plot convergence curve of lowerbound during training phase.

```
figure
plot(mdl.out.lbBar,'LineWidth',2)
title('Lowerbound of Variational Approximation','FontSize',20)
xlabel('Iterations','FontSize',14,'FontWeight','bold')
ylabel('Lowerbound','FontSize',14,'FontWeight','bold')
grid on
```

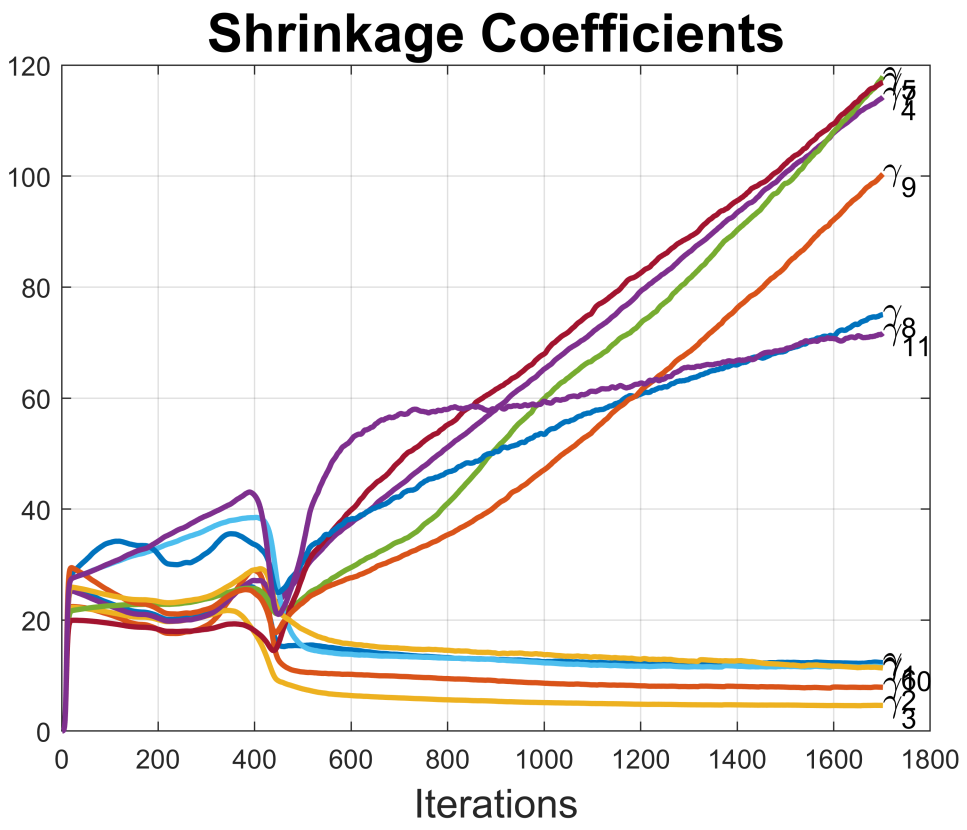


Variable selection

Now plot Shrinkage coefficients to show if adatptive group Lasso can perform variable selection efficiently.

Specify shrinkage option to plot shrinkage coefficient from training phase

```
deepGLMplot('Shrinkage',mdl.out.shrinkage,...     % Plot Shrinkage coefficients
            'Title','Shrinkage Coefficients',...
            'Xlabel','Iterations',...
            'LineWidth',2);
```



The shrinkage coefficients of variables **4,5,7,8,9,11** significantly higher than the others at the end of training phase.

So these variables are not important in the model.

Now let's check if linear regression model gives us the same results on variable selection

Now let's check if linear regression model gives us the same results on variable selection

```
mdlOLS = fitglm(X,y)
```

```
mdlOLS = 
Generalized linear regression model:
    y ~ 1 + x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 + x9 + x10 + x11
    Distribution = Normal

Estimated Coefficients:
                   Estimate        SE         tStat        pValue  
                   _________    ________    ________    __________

    (Intercept)       1.2083    0.016228      74.459             0
    x1              -0.20185    0.019185     -10.521    1.7685e-24
    x2               0.27858    0.017178      16.217    5.3457e-52
    x3               0.63694    0.032713       19.47    1.3714e-70
    x4              0.010944    0.017371     0.63004       0.52883
    x5             -0.017643    0.023582    -0.74813       0.45458
    x6              -0.21175    0.017007     -12.451    6.4572e-33
    x7              0.014436    0.019354     0.74594        0.4559
    x8             -0.014551    0.026035     -0.5589       0.57637
    x9           -0.022722    0.025423    -0.89377       0.37169
    x10            -0.1197    0.018063     -6.6267    5.9474e-11
    x11           0.017864    0.023284     0.76722       0.44316


900 observations, 888 error degrees of freedom
Estimated Dispersion: 0.237
F-statistic vs. constant model: 229, p-value = 1.49e-250
```

p-values of variables 4,5,7,8,9,11 are much higher than the others

This variable selection result perfectly matches with **deepGLM**

Prediction

The next step is to make prediction from trained model. deepGLM provide point and interval estimation for unseen data.

In order to make point estimation for new data,
use **deepGLMpredict** function. **deepGLMpredict** outputs an array of point estimations of observations in test data

```
Pred1 = deepGLMpredict(mdl,X_test);
```

If responses for test data (ytest) is specified (for model evaluation purpose) then we can check PPS and MSE on test set. Specify test data with option **'ytest'.** The output of **deepGLMpredict** is a structure including:

- An array of prediction values for each observation in test data
- PSS loss of test data
- MSE of test data

```
Pred2 = deepGLMpredict(mdl,X_test,'ytest',y_test);
disp(['PPS on test set using deepGLM is: ',num2str(Pred2.pps)])
```

```
PPS on test set using deepGLM is: -0.17324
```

```
disp(['MSE on test set using deepGLM is: ',num2str(Pred2.mse)])
```

```
MSE on test set using deepGLM is: 0.25196
```

deepGLM also provides prediction interval option.

In order to make interval estimation, specify the interval (within how many std) for option **'Interval'** in the function **deepGLMpredict.**

Provide sampe size using **'Nsample'** options.

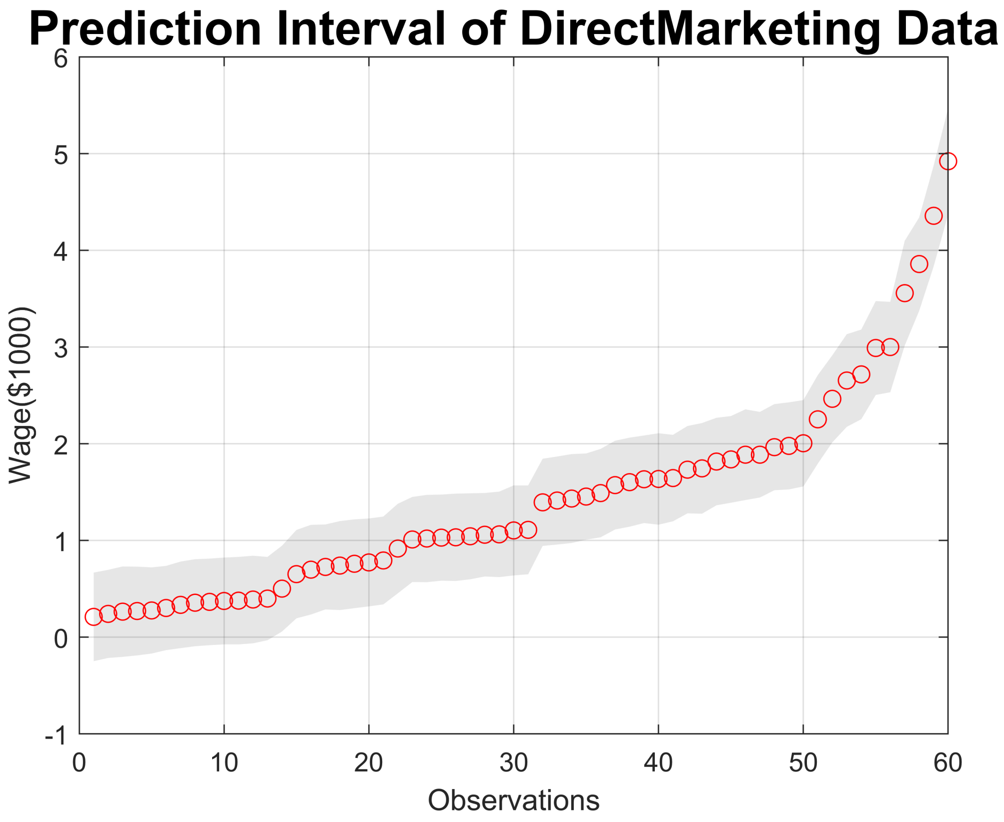
```
Pred4 = deepGLMpredict(mdl,X_test,...
                       'Interval',1,...  % Give prediction interval within 1 std from mean for
                       'Nsample',1000);  % Number of samples generated from VB Posterior distr
```

```
% Calculate how many percent of times that the prediction invertals are correct
accuracy = (y_test<Pred4.interval(:,2) & y_test>Pred4.interval(:,1));
disp(['Prediction Interval accuracy: ',num2str(sum(accuracy)/length(accuracy))]);
```

```
Prediction Interval accuracy: 0.78
```

Now we can plot the prediction interval for a subset of unseen data points. Specify **'interval'** option for **deepGLMplot**

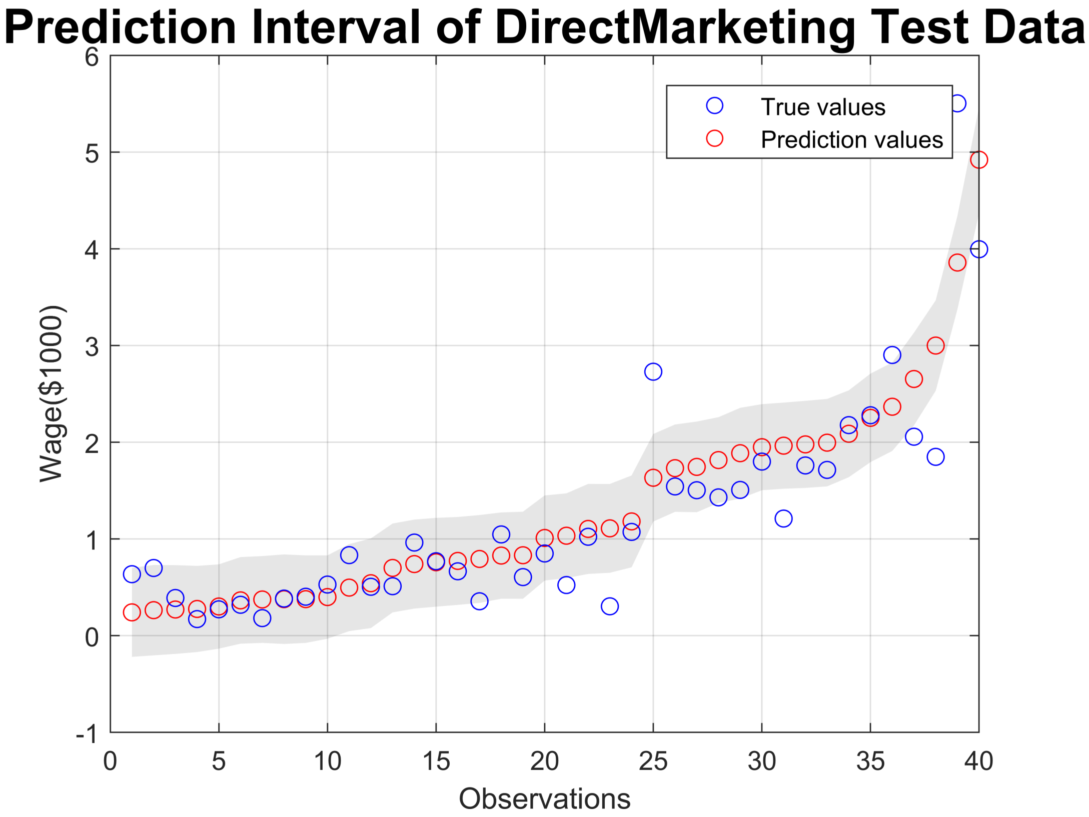
```
figure
deepGLMplot('Interval',...                          % Plot prediction intervals
             Pred4,...
            'Title','Prediction Interval of DirectMarketing Data',...
            'Xlabel','Observations',...
            'Ylabel','Wage($1000)',...
            'Nsample',60);                          % Randomly pick 60 observations to plot
```



If we want to plot the true responses, the specify true response vector for the option **'ytest'**

```
figure
deepGLMplot('Interval',...
             Pred4,...
            'ytest',y_test,...                       % Plot true responses
            'Title','Prediction Interval of DirectMarketing Test Data',...
```

```
              'Xlabel','Observations',...
              'Ylabel','Wage($1000)',...
              'Nsample',40);                    % Randomly pick 40 new observations to plot
```


Prediction Interval of DirectMarketing Test Data
True values
Prediction values
Wage($1000)
Observations