Pearson Plot Shows Relations between Many Variables in One Figure

Sven Ahlinder

***Abstract*— Data analysis is much about correlations. Correlations between features and responses are good. Correlations between features are bad. Visualization of relations between many variables is a general problem. The Pearson Plot introduced here shows the relation of many variables to two chosen. This may be a valuable tool in data analysis for finding correlations in data**

I. INTRODUCTION

1. Spurious correlations can make the data analyst draw completely wrong conclusions. Spurious correlations occurs when features in in-data table are correlated.
2. In feature selection it is good to know which features that correlates with the response and which features that do not correlate
3. When optimizing, there is valuable to know which responses that correlate and which responses that are contradictive

All those three topics can be illustrated and solved using the "Pearson plot" that is presented here. The plot is made as:

1. Concatenating features and responses into one single table
2. Choice of two columns in the joint table of features and responses by user
3. Constructing a plane from the points of the two selections and center of gravity of dataset
4. Projection of all features and responses on this plane
5. Enable new choice of plane to investigate new correlation structures

Manuscript received May 25, 2020
Sven Ahlinder, Volvo Powertrain Strategic Development

EXAMPLE

Making vehicles is a struggle on keeping efficiency high and emissions low. Table 1 shows an experimental series to simultaneously lower emissions and increase efficiency. The parameters studies are Timing, Pressure, Temperature and EGR. It is hard to draw conclusions from the table.

no	Timing	Pressure	Temp	EGR	Emission	Efficency
1	0.496267	0.497315	0.452166	0.495597	0.314604	0.170966
2	0.49718	0.500066	0.701001	0.063589	0.42917	0.942913
3	0.500054	0.499133	0.33724	0.831784	0.27172	0.034292
4	0.499652	0.496829	0.359611	0.507422	0.458381	0.111939
5	0.500484	0.49779	0.569397	0.497987	0.262947	0.206889
6	0	0.496942	0.313867	0.477072	0.76062	0.725346
7	1	0.496832	0.536669	0.463623	0.115728	0.016759
8	0.988851	1	0.866071	0	0.44705	0.438052
9	0.992292	0.996232	0.15048	0.844102	0.573659	0.184188
10	0.501928	0.497892	0.444728	0.508729	0.320834	0.168371
11	0.499398	0.997525	0.495412	0.421407	0.497353	0.383652
12	0.008786	0.993533	0.179283	0.812555	1	0.265919
13	0.009178	0.997965	0.142472	0.776357	0.887597	0.870656
14	0.993916	0.994876	0.77677	0.058883	0.308194	1
15	0.99394	0.99375	0.6152	0.866632	0.212659	0.016556
16	0.504486	0.489303	0.559931	0.457519	0.333646	0.099465
17	0.498203	0.497365	0.340255	0.38956	0.442225	0.444158
18	0.005005	0.004091	0	0.855589	0.841242	0.871663
19	0.001813	0.001492	0.142376	0.85889	0.447862	0.499877
20	0.01056	0.001033	0.336767	1	0.41197	0.103822
21	0.996705	0.02977	0.52822	0.025553	0.501658	0.849854
22	0.994905	0.007394	0.381396	0.869653	0.153467	0
23	0.990768	0.168078	1	0.00532	0.079918	0.428636
24	0.985331	0.001657	0.33826	0.826287	0	0.046467
25	0.498826	0	0.474582	0.451882	0.199182	0.107223

Table 1 An engine experiment. All data scaled between 0 and 1

Table 2 shows the same values, but projected on the plane spanned by Emission and Efficiency. We see that

- "Efficiency" lies on x= 1, y=0
- "Emission" is strong in x but reasonable strong in y too
- We should increase x and decrease y
- This is done by increasing "Temp"

	x	y
Timing	-0.32	-0.95
Pressure	0.99	0.10
Temp	0.28	-0.96
EGR	-0.62	-0.78
Emission	0.85	0.52
Efficency	1.00	0.00

Table 2 Projection on Emission and Efficiency

Figure 1 shows the Matlab tool FitCorrSel.m. The diagram is the plot of Table 2 with origin in the middle of the diagram. We can see from the diagram that:
"Efficency" and "Emission" are heavily correlated
Increasing "Pressure" will increase both "Efficency" and "Emission".
Increasing "Timing" and "Temp" will reduce "Emission" but not effect "Efficiency" since "Emission" do correlate with the "y" axis and "Efficency" does not.

The Matlab tool FitCorrSel.m provides buttons in top for
"Turn" to select which two variables that shoud span the projection plane
"Select in/out" to select included variables
"Fit in/out" to provide regression coefficients for selected variables

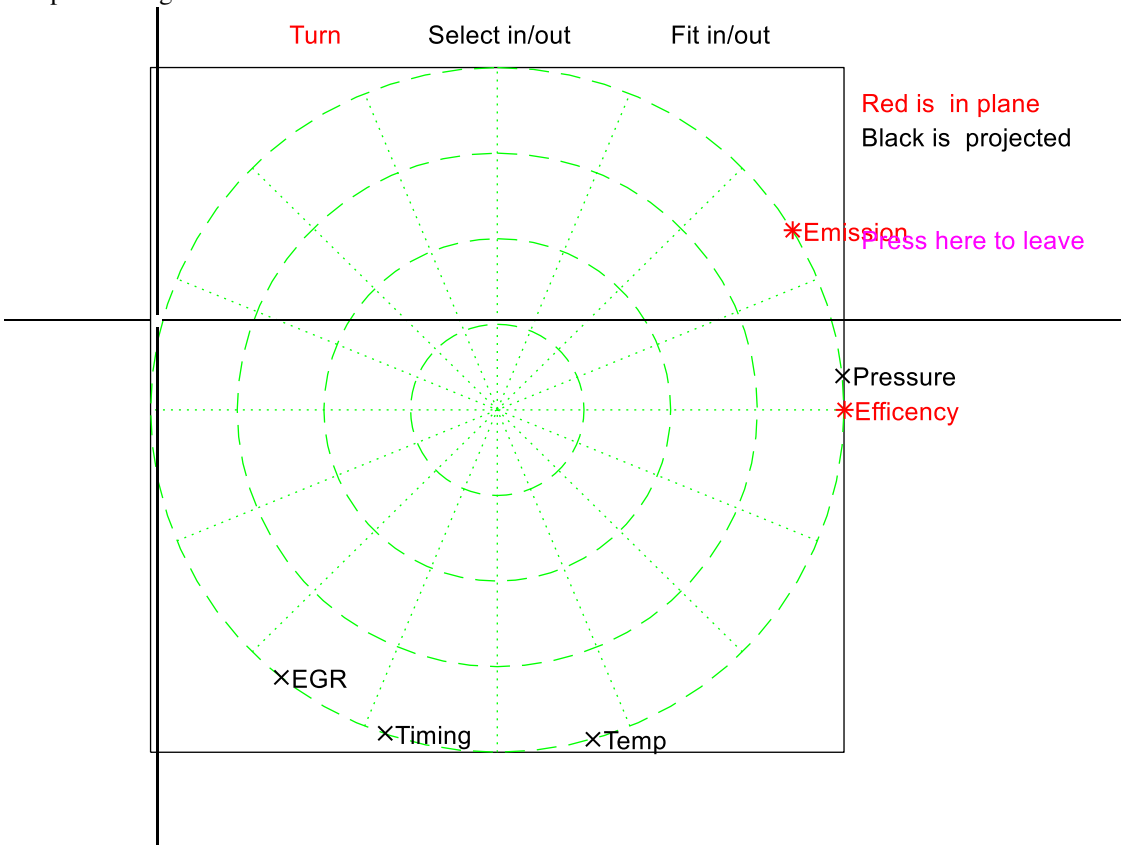

Figure 1 Matlab function FitCorrSel

MATHEMATICS

The "in-data" is a matrix with variables in columns and examples in rows, Table 1.
The "projection-variables" are two columns from"in-data".
The "projection-matrix"=
projection-variables* inv(projection-variables'* projection-variables)* projection-variables
The "projection"= projection-matrix* in-data
The "Correlation-matrix"= "pearson´s-correlation-coefficents" of projection.
The "preliminary-plot-matrix" are the two columns in correlation matrix corresponding to projection-variables
The first column, "x" of "plot matrix" is the column in preliminary-plot-matrix that contains a value exact "1".
The second column, "y", of "plot matrix" is
"sign of remaining column of preliminary-plot-matrix"
multiplied with the corresponding value of
square root of (1- " x with every value squared")
Appendix, the code in program language Matlab

```
function PlotMatrix= PearsonplotTurn( PlottingMatrix, PlottingLabeling)
% Projects column 3 and following onto plane spanned by column 1 and 2
% Projection
% https://en.wikipedia.org/wiki/Projection_(linear_algebra)
Projector=   PlottingMatrix( : , 1: 2)+ 1e-6;
Projector= Projector* inv( Projector'* Projector)* Projector';
Projection= Projector* PlottingMatrix;
```

```
% Pearson's
% https://en.wikipedia.org/wiki/Correlation_coefficient
CorrelationMatrix= corrcoef( Projection+ 1e-6);
% Correlations to spanninng variables
PlotMatrix= CorrelationMatrix( :, 1: 2);
% Correlation of spanning variable 2 to spanninng variable 1
PlotMatrix( :, 2)= sign( PlotMatrix( : ,2)).* sqrt( 1 -PlotMatrix( :, 1).^ 2);

% Projection distance according to each variables correlation to projection
Corr= corrcoef([ Projection, PlottingMatrix]);
n= size( Projection, 2);
for i= 1: n;
    PlotMatrix( i, :)= PlotMatrix( i, :)* Corr (n+ i, i);
end
```