REGRESSION

AUTHORS:
ANGAD GADRE
HARISH MANGALAMPALLI
RAJARAM RABINDRANATH

Contents

EXPERIMENTS

Problem 1: Experiment with Linear Regression

Implement ordinary least squares method to estimate regression parameters by minimizing the squared loss.

Calculate and report the RSE for training and test data for two cases: first, without using an intercept (or bias) term, and second with using an intercept.

	Training Dataset	Test Dataset	% Increase in error when testing model
Without Intercept	2149.9	4621.2	114.95%
With Intercept	727.52	861.14	18.37%
% decrease in error post intercept	66%	81%	

Which one is better?

We find that the error is lower when an intercept (or bias term) in used when compared to the experiment where an intercept is not used. From the results it is evident that the addition of intercept to the regression model helps reduce the errors with respect to predictions, both, in case of training and test dataset.

Problem 2: Experiment with Ridge Regression

Implement parameter estimation for ridge regression by minimizing the regularized squared loss as follows:

$$J(w) = \frac{1}{N}\sum_{i=1}^{N}(y_i - w^T x_i)^2 + \lambda w^T w$$

Calculate and report the RSE for training and test data using ridge regression parameters. Use data with intercept.

- Error for training data **770.376** at $\lambda_{optimal}$
- Error for test data **755.1602** at $\lambda_{optimal}$

Compare the relative magnitudes of weights learnt using OLE (Problem 1) and weights learnt using ridge regression.

The weights learnt using ridge regression are significantly lower in magnitude (about 55 times lower) as compared to the weights learnt using the Ordinary Least Squares (OLE) approach.

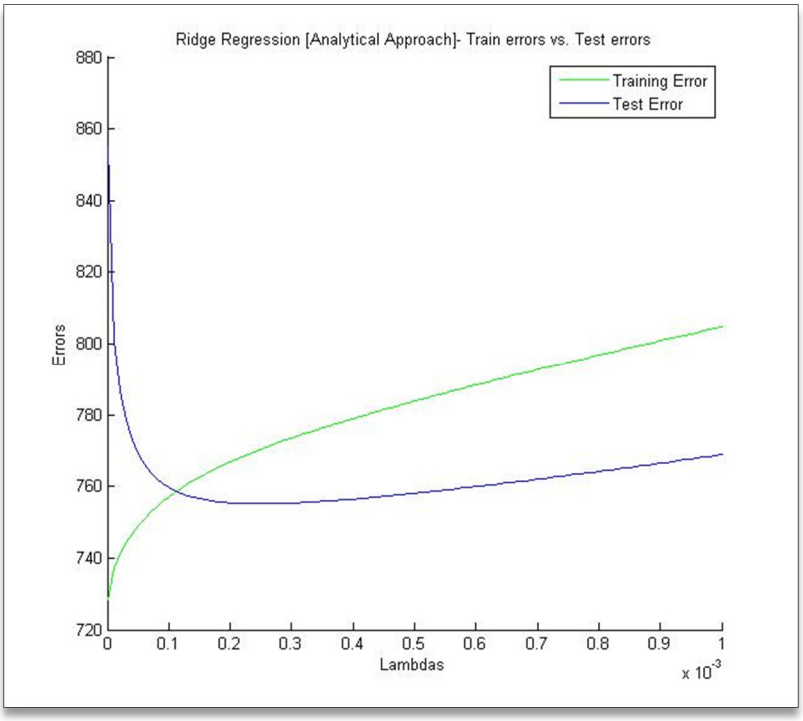
Average of the weights (absolute values):

- Ordinary Least Squares: **4408.8**
- Ridge Regression : **80.4116** at $\lambda_{optimal}$

Compare the two approaches in terms of errors on train and test data.

The first approach (OLE) gives lower training error when compared to ridge regression. This is because it better fits the training data (over-fit) and fails to generalize well. However, ridge regression gives lower error on the test data (for optimal value of $\boldsymbol{\lambda}$) when compared to the OLE approach. As mentioned above, this is because we are regularizing the weights (reducing their magnitude) in order to make a better generalization.

Plot the errors on train and test data for different values of $\boldsymbol{\lambda}$. Vary $\boldsymbol{\lambda}$ from 0 (no regularization) to 0.5 in steps of 0.001.



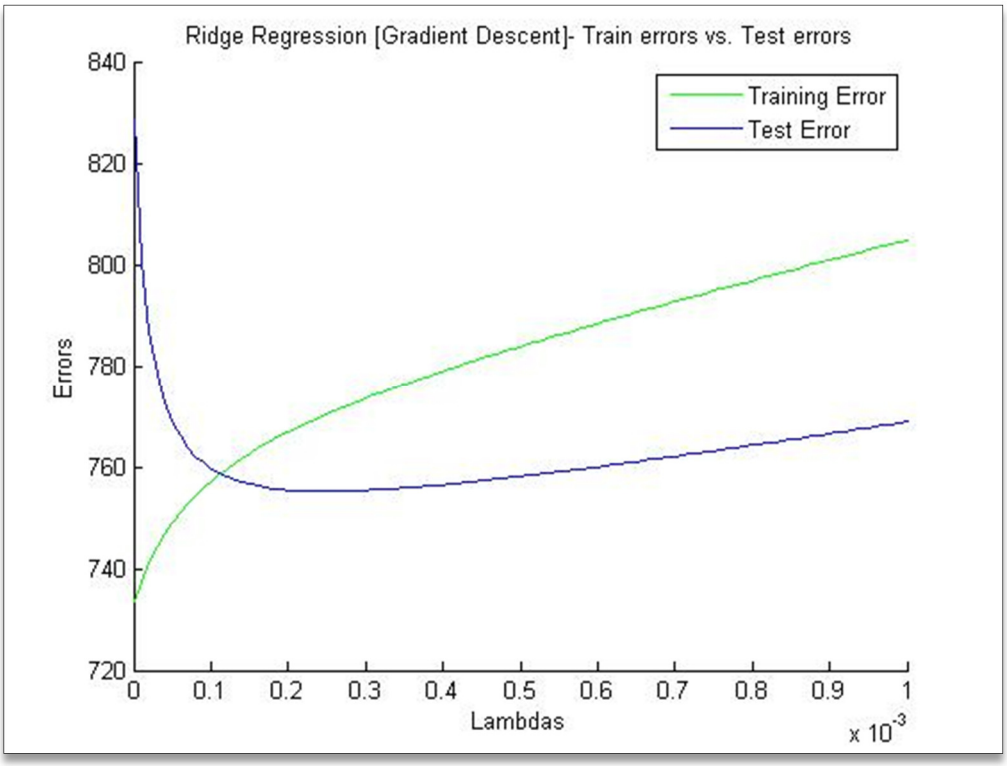
What is the optimal value for $\boldsymbol{\lambda}$ and why?

The optimal value of $\boldsymbol{\lambda}$ is 2.5×10^{-4}. It is obtained from the graph of train errors and test errors against $\boldsymbol{\lambda}$ where the test error is minimum. The curve for training error against $\boldsymbol{\lambda}$ is a monotonically increasing function. This is because the optimal fit on the training data (over-fit) occurs when regularization is not used, i.e., when we fit a line that closely follows the trend in the training data. However, our objective is to obtain the best generalization of the regression line from the training data. This can only be done if we validate our fit against a validation set. In the present case, since we do not have a validation set at our disposal, we use the test set to validate our generalization, i.e., the optimal value of the

regularization coefficient ($\boldsymbol{\lambda}$) and the corresponding weights. As we keep incrementing the impact of the regularization term and as a consequence bring the weights down, we find the errors on the test set to decrease. However; beyond a point any increase in the regularization term negatively impinges on the predictions of both training and test data, as a continual increase in the value of $\boldsymbol{\lambda}$ ignores the learnings from the training data (in essence we lose what we are learning). The value of $\boldsymbol{\lambda}$ that gives the least error on the test set is the best generalization of our model and hence we use this value as the optimal value of $\boldsymbol{\lambda}$.

Problem 3: Using Gradient Descent for Ridge Regression Learning

Plot the errors on train and test data obtained by using the gradient descent based learning by varying the regularization parameter $\boldsymbol{\lambda}$.



Optimal value of $\boldsymbol{\lambda}$:

- $\mathbf{2.5x10^{-4}}$ with the corresponding error value of **755.1602** for test data

Compare with the results obtained in Problem 2

The results obtained using gradient descent are the almost similar to those obtained in problem 2. In fact, the same value of λ is observed as the optimal value in both cases and so is the corresponding error value with respect to test data predictions.

Given that the optimal value of λ and the corresponding error values are the same in both cases i.e Problem #2 & Problem #3, the only other metric that could be used to compare these two techniques is learning time. Following are the execution times:

CPU time taken:

- Ridge Regression—Analytical Approach : **0.0938 seconds**
- Ridge Regression— Gradient Descent: **358.3125 seconds**

The 'gradient descent approach' takes **3819.9626x** more time than the 'analytical approach' to arrive at the same optimal value of lambda. Add to that the fact that the number of examples used for training were only 242. Clearly the analytical approach followed in problem #2 is a better method of finding the optimal value of λ.

Note that this comparison holds for our experiments using MATLAB where the inverse computations are handled elegantly using approximations internally. However, if we were to develop and use our own un-optimized code for computing inverse of matrices, we would find the gradient descent approach to perform better (as expected).

Problem 4: Non-Linear Regression

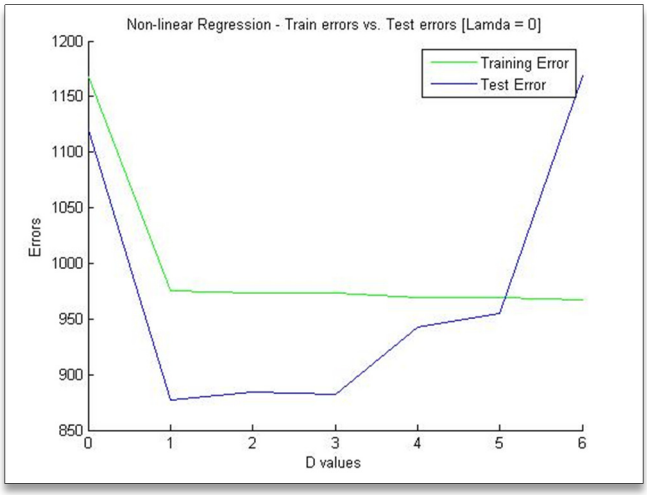
In this problem we will investigate the impact of using higher order polynomials for the input features. For this problem use the third variable as the only input variable:

$$\text{x\_train} = \text{x\_train}(:,3);$$

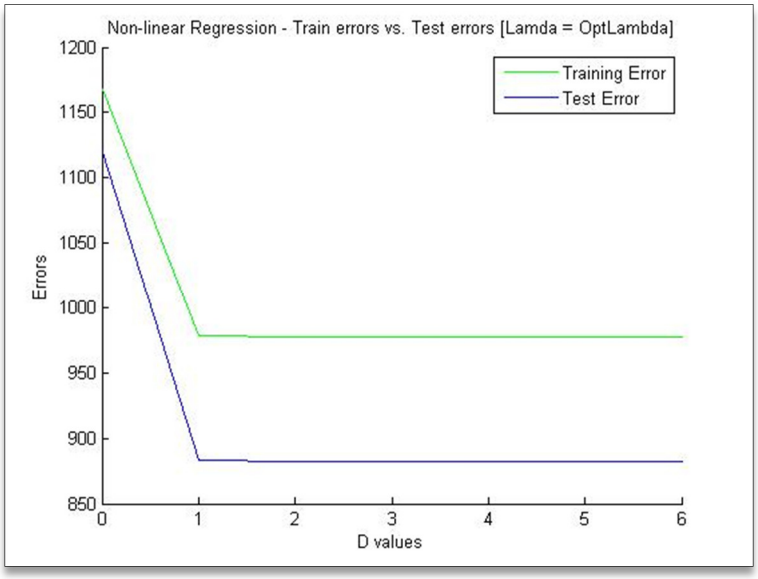
$$\text{x\_test} = \text{x\_test}(:,3);$$

Using λ = 0 and the optimal value of λ found in Problem 2, train ridge regression weights using the non-linear mapping of the data. Vary d from 0 to 6. Compute the errors on train and test data.

Non-linear Regression for d [0…6] & $\lambda = 0$ [no regularization]



Non-linear Regression for d [0…6] & $\lambda = 2.5x10^{-4}$



Compare the results for both values of λ.

When the OLE approach is used (no regularization, i.e., λ = 0; refer equation 3 below), we observe that the training error decreases gradually as the value of d increases. This can be attributed to the fact that as the complexity of the model (or the number of non-linear terms in the augmented feature space) increases, the model fits better to the training data. An obvious repercussion of increasing *d* is that the model fails to generalize well. Thus, the test error increases as d increases.

For ridge regression (i.e, λ > 0; refer Equation 3 below), we observe the training and testing errors to decrease initially from d = 0 and then level off. The testing errors are always lower than the training errors showing that the model obtained using regularization generalizes well for all values of d. The errors with respect to the training data does not increase beyond a point because the non-zero λ value helps keep the weight in check and does not let them become complex.

Note: That at d = 0, we are not using the training data to generate our model (high bias), because of which both the training errors and testing errors are very high as compared to errors it fits using higher values of d.

What is the optimal value of d in terms of test error in each setting?

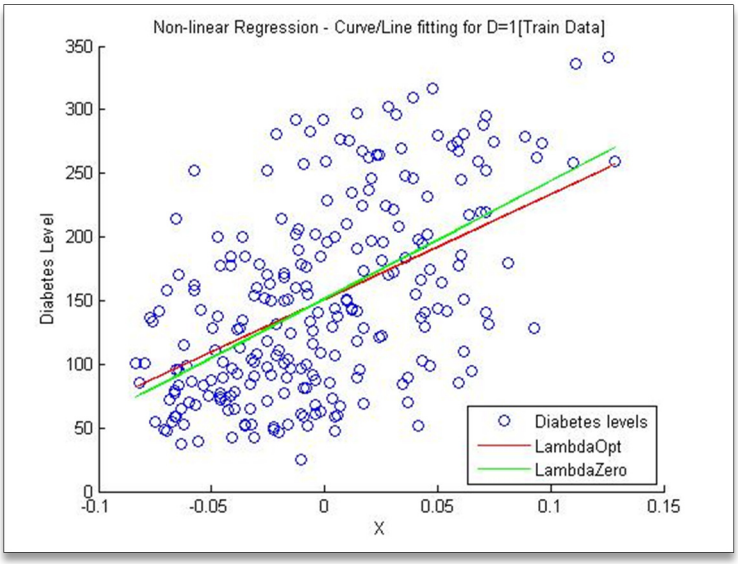
Optimal Value of d and the corresponding error on test dataset:

- No regularization: d = 1 → ERROR: 876.9304
- λ = optimal λ found in problem 2: d = 3 → ERROR: 882.7408

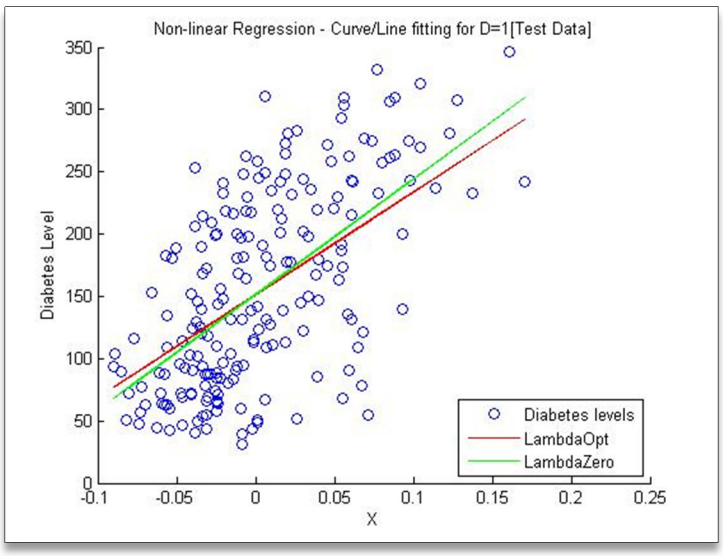
From the above listing it is clear that the least error when predicting Diabetes levels is minimum for the no regularization and d=1 setting. We therefore proceed to plot the regression lines for $\lambda = 0$ and $\lambda =$ optimal λ value found in problem 2, while keeping D constant at 1

Plotting the curve for the optimal value of D for both values of λ and compare.

Fitting the Curve for the best value of d = 1[Train Data]



Fitting the Curve for the best value of d = 1[Test Data]



Following are the weight vectors that were computed:

- Weight vector when $\boldsymbol{\lambda} = 0$
  - $W_0 = 151.4464$
  - $W_1 = 927.1199$
- Weight vector when $\boldsymbol{\lambda = 2.5x10^{-4}}$
  - $W_0 = 151.1615$
  - $W_1 = 824.1032$

Following are the Root Squared Errors:

ERRORS WHEN D=1 [Optimal]			
Lambdas	**Train Data**	**Test Data**	**% decrease**
Lambda=0	975.3366	876.9304	10.09%
Lambda=optimal	977.9675	882.7724	9.73%
% increase	0.27%	0.67%	

From the plots, weights and the errors we can see that there is a perceptible change in the outcome between no regularization and using regularization. We see that the weights do get affected when using the regularization term, which is along the expected lines. The decrease in W_0 is negligible while the decrease in W_1 is reasonable. No regularization outperforms the one with regularization since the lambda chosen is the one that was optimal for Ridge Regression and not for non-linear regression.

Problem 5: Interpreting Results

Using the results obtained for previous 4 problems, make final recommendations for anyone using regression for predicting diabetes level using the input features.

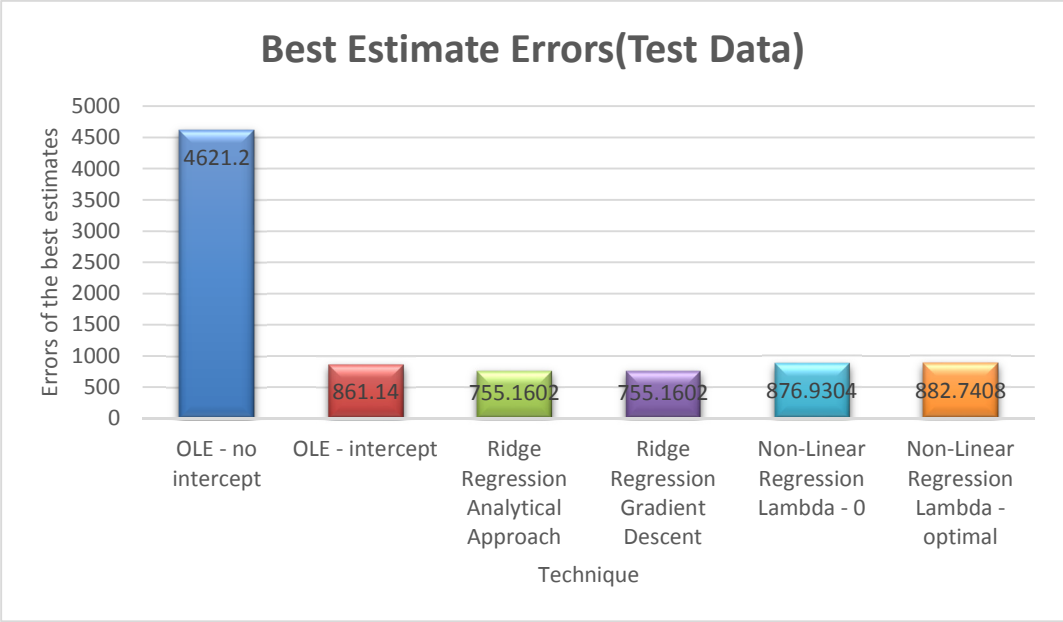
Compare the various approaches in terms of training and testing error. What metric should be used to choose the best setting?

Experiment	Time Taken(Seconds)	Best Error(Test Data)
OLE - no intercept	0	4621.2
OLE – intercept	0.0313	861.14
Ridge Regression Analytical Approach	0.0938	755.1602
Ridge Regression Gradient Descent	358.3215	755.1602
Non-Linear Regression Lambda – 0	0.0156	876.9304
Non-Linear Regression Lambda – optimal	0.0313	882.7408

If we were to rank the above listed techniques by Best Error we shall find the following ranking:

1. Ridge Regression using Analytical Approach
2. Ridge Regression using Gradient Descent
3. OLE-intercept
4. Non-Linear Regression
5. OLE-no Intercept

Following is a graph showing the prediction errors (RSE) of the best estimates of each technique:



Recommendations & Suggestions:

From the results of the 1st problem, we learnt that using an intercept (or bias) term produces a better generalization than the model that does not use an intercept. Ridge regression with the optimal value of λ produces better generalization with lower test error. Ridge regression using gradient descent for optimization produces more or less the same results. Non-linear regression shows no significant increase in performance (decrease in error) with increase in the value of d, i.e., the number of dimensions onto which the data is mapped. So, from the results we obtained for the 4 problems, we conclude that the best method for regression for predicting diabetes levels would be ridge regression.

In our opinion, the best metric to choose the right setting for regression would be the Root Square Error (RSE) as used in this assignment.

However the ranking above are very specific to the dataset that we have operated on and therefore must not be taken as absolute truths but rather as a guide which should be tempered by one's domain knowledge. The following section lists our learnings and inferences.

Learnings & Inferences:

- **OLE – without Intercept**
  - The manifestations of erroneously omitting the intercept would cause the predictor line ($w^t * X$) to
    - Have a steeper slope &
    - Be biased towards the Y values corresponding to the large values of each feature $\{x_1, x_2, \dots, x_D\}$
    - Resulting in embarrassingly large errors; evident from our results

- Must be used only in cases where the user (who we presume is acutely aware of the nature of the training data) has a logical reason omit intercept
  - E.g. Not having an intercept while predicting the age of a tree or in economic models where it is illogical to have a non-zero value of Y when all features values are zero [1]
  - Despite the above logical reasoning, omitting the intercept may not bode well for a predictor as it may not capture the **accurate linear trend**
- Even in scenarios where it is known that the data being modeled on is from a generator that does not have an intercept; it can be shown that incorporating the intercept term results in better predictions [1]

➢ **OLE – with Intercept**

- Is a better predictor with respect to *'OLE-no intercept'* due to the reasons listed above
- Must be used even when it is illogical to have the model predicting a non-zero value for Y when all feature values are 0; as lack of intercept shall impact the slope of the predictor line thereby adversely impacting prediction accuracy
- The weights learned to fit the predictor line is begotten by minimizing the loss function with respect to the weight vector; this shall cause over-fitting w.r.t to the training data resulting in greater predictions errors vis-à-vis test data
- The equation used to estimate the weights for the OLE experiments :
  - $\boldsymbol{W_{mle} = \{X^T X\}^{-1} * X^T y}$ **-- Equation 1**
  - Note : In some cases $\{X^T X\}^{-1}$ may not exist

➢ **Ridge Regression Analytical Approach**

- To counter the problem of over-fitting [as seen in the case of OLE-with/out intercept] we use ridge regression which penalizes the large weights for wrong predictions by introducing a regularization term
  - Fortuitously this also solves the problem of possible non-existence of $\{X^T X\}^{-1}$
- The loss function for ridge regression including the regularization parameter:
  - $\boldsymbol{J(w) = \frac{1}{N}\sum_{i=1}^{N}(y_i - w^T x_i)^2 + \lambda w^T w}$ **-- Equation 2**
- The best weights are derived from **Equation 1** to be:
  - $\boldsymbol{W_{mle} = \left[\left(XX^T\right) + N\lambda I_D\right]^{-1} * X^T y}$ **-- Equation 3**
- As a consequence we get lesser prediction errors vis-à-vis the test data
- Ridge regression performs especially well in cases wherein we have small number of examples and large number of features
  - Such a dataset shall have very little variation to present the learner with, resulting in over-fitting by OLE, Ridge Regression handles this case well

➢ **Ridge Regression Gradient Descent**

- Computing an inverse in **Equation 2**, could be computationally complex
- We therefore could use gradient descent technique to find the right W and reduce the computational complexity

- The loss function used is **Equation 2**
- The error gradient calculated using Equation 2, turns out to be
  - $Error_{grad} = \frac{2[(X^T X)w - X^T y]}{N} + 2\lambda w$ **-- Equation 4**

➢ **Non-Linear Regression**

- All the aforementioned techniques assumed the existence of a linear relationship between the dependent variable ,'Y', and all the independent variables, features
  - This may not be true all the time
- When the relationship between dependent and independent variables is non-linear, fitting a non-linear curve shall give us better accuracy

REFERENCES

[1] http://www.talkstats.com/showthread.php/18336-Use-of-Intercept-Term-in-OLS-Regression

[2] http://statistiksoftware.blogspot.com/2013/01/why-we-need-intercept.html

[3] http://statistiksoftware.blogspot.com/2013/01/a-discussion-on-non-hierarchical.html