Machine Learning Assignment 2

Manas Gaur, Wright State University 10/11/2017

1 Introduction

In this, we will be exploring different types of regression functions in machine learning. Such as Logistic regression in one variable, logistic regression with multiple variables, Regularization, and Feature Scaling. Our analysis and modeling will be using flu dataset. The flow of the assignment is organized as follows; Firstly, we will analyze the efficiency of the logistic regression with one variable using confusion matrix values and root means square. Secondly, we will extend this model to adapt to multiple variables in the dataset and finally we will modify the cost function of the model with regularization and feature scaling on the dataset. The dataset provided has 417 data points and 18 features (we consider Flu as the target variable in this assignment). In 2011, using the self-report survey of 410 high school students in the mid-west, the data was collected. In this project we will assess impact **Risk**, **KnowItrans**, **HndWshQUal** and **Gender** on Flu.We enlisted the descriptive values of the features in table 1.

Table 1: Features and Data type

Feature Names	**Data Type**	**Type of Variable**	**Range**
HndWshQual	Likert	Independent	[1.0, 5.0]
Risk	Real number	Independent	[-1.453,1.453]
KnowlTrans	Real Number	Independent	[-1.393, 1.393]
Flu	Binary	Dependent	0.0,1.0
Gender	Binary	Independent	0.0,1.0

2 Data Visualization

In this section, we will discuss the distribution the data points in the coordinate plane. We analyze the correlation between the independent variable with the target variable by plotting the cumulative sum plot.

In the figure 1, we illustrate that the observations of Flu and Risk pair are dichotomous with a majority of the observations having Flu value 0. Such distribution can be seen to be solvable using linear regression model but we will observe how non-linear model understands this scenario.

In the figure 2, we illustrate, how the cumulative sum over all the observation of the independent variable varies with the target variable.The cumulative plot is used to show the number of

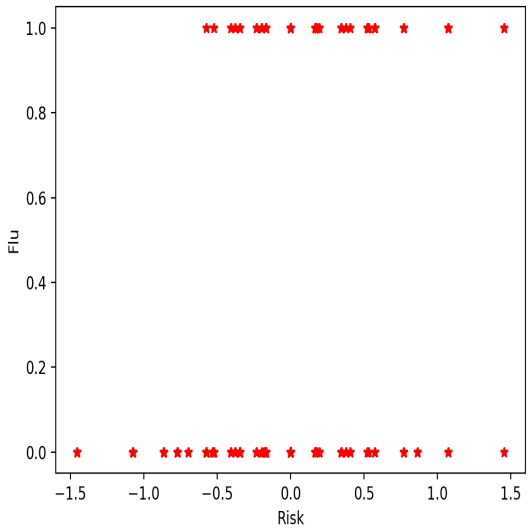
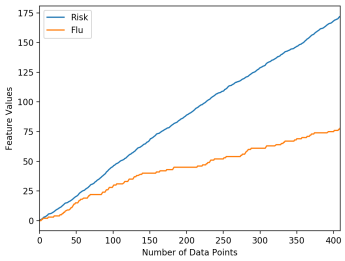
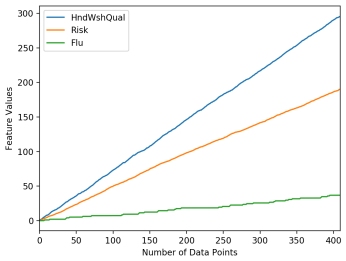


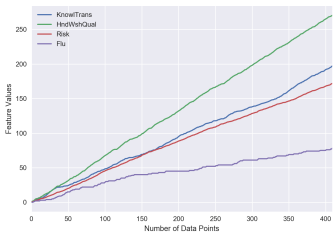
Figure 1: Plot of Risk versus Flu



(a) Risk and Flu



(b) {Risk, HndWshQual} Flu



(c) {Risk, HndWshQual, KnowlTrans} Flu

Figure 2: Figure shows the cumulative sum plot of the dataset which is enhanced by adding more features. (a).cummulative sum of flu values is not a straight line. There is a high value of zeros as compared to ones. Values of Risk is negative most of the times in the dataset. (b). HndWshQual has a high gradient as compared to Risk and Flu. (c). Variation in HndWshQual is high as compared to KnowlTrans. This gradient in the feature values is reflected in the model efficiency (illustrated in subsequent sections).

Table 2: Contingency table for Flu, HndWshQual and Risk

	HndWshQual	Risk	Flu
HndWshQual	1.0	-0.06	-0.04
Risk	-0.06	1.0	0.08
Flu	-0.04	0.08	1.0

Table 3: Contingency table for Flu and Risk

	Risk	Flu
Risk	1.0	0.344
Flu	0.344	1.0

observations that lie above (or below) a particular value in a data set. We analyze the plot by generating a contingency table showing the Pearson correlation value of the attribute. In table 3, Risk and Flu are positively correlated, while HndWshQual shows a negative correlation with Risk and Flu (though the correlation coefficient is very small).

Note: We have made the cumulative sum plot of $\{Risk, Flu\}$ and $\{Risk, HndWshQual, Flu\}$. This plot can be easily created using pandas. A function in the name *analyzing_features* is provided in the program which takes a pandas data frame an argument. The plot provides a fairly good understanding of the region where the values of the features lie in the dataset.

Figure 3 plot Flu as a function of HndWshQual and mean of Risk. It illustrates that with the increase in the quality of Hand-Wash (HndWshQual), the value of risk decreases. Moreover, we see Risk value of 0.5 is behaving as a threshold for classifying Flu as 0.0 or 1.0. A deep insight obtained from variance lines shows that this two feature variance lies in the range of [0.2,0.7] which is less as compared to figure 4. Moreover, we observe that the influence of Risk on Flu is more than HndWshQual on Flu, this is attributed to a high correlation between Risk and Flu. An understanding of this phenomenon is visible in subsequent sections where we observe the change in efficiency of the model with the addition of a new feature to the dataset.

Table 4: Contingency table for Flu, Risk, HndWshQual and KnowlTrans

	KnowlTrans	HndWshQual	Risk	Flu
KnowlTrans	1.000000	0.043471	-0.023555	-0.059531
HndWshQual	0.043471	1.000000	-0.036527	-0.154261
Risk	-0.023555	-0.036527	1.000000	0.342740
Flu	-0.059531	-0.154261	0.342740	1.000000

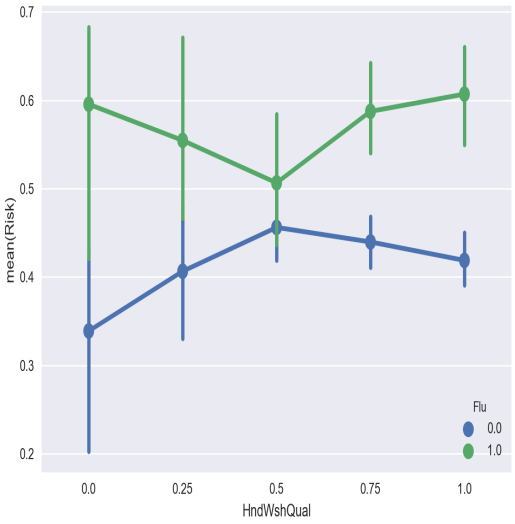


Figure 3: Plot of HndWshQual over Risk with points segmented by Flu labels

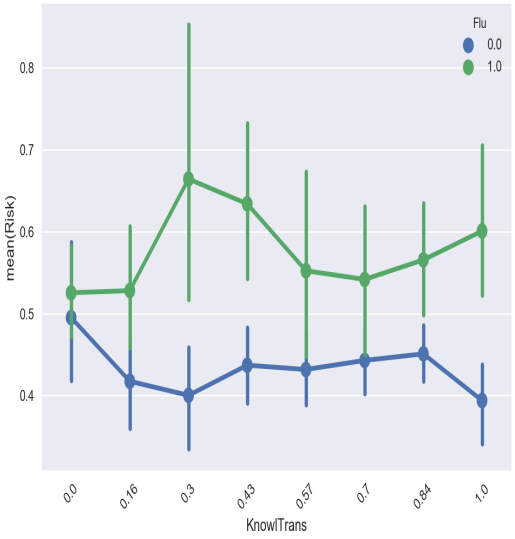


Figure 4: Plot of KnowlTrans over Risk with points segmented by Flu labels

Table 5: Cross Validation Fscore results tables. LogR1V : Logistic Regression with One Variable,

%age Training	%age Testing	Precision	Recall	F-score
20	80	0.20	1.0	0.33
30	70	0.21	1.0	0.35
40	60	0.23	1.0	0.37
50	50	0.26	1.0	0.41
60	40	0.26	1.0	0.41
70	30	0.26	1.0	0.41
80	20	0.27	1.0	0.42
90	10	0.30	1.0	0.46

Table 6: Confusion matrix of the logistic regression with one variable. Threshold kept very low : 0.2

	Predicted-True	Predicted-False
Actual-True	12	0
Actual-False	28	1

In the figure 4, we illustrate the trend in flu using KnowlTrans and mean Risk values. An observation is a threshold, which is 0.5, similar to figure 3. Moreover, KnowlTrans value between [0.5, 0.55] and Risk value between [0.6, 0.7] has abnormal high variance in comparison to neighborhood data points. Moreover, there is minuscule (negative) correlation between KnowlTrans and {Flu, Risk} (refer table 4).

3 Logistic Regression with One Variable

In this section we develop logistic regression over a dataset with one independent variable and one target variable. The independent variable is *Risk* and dependent variable is *Flu*. Variable *Risk* is under the domain of real numbers and *Flu* is binary. Since both features have a domain, we need to normalize the *Risk* and keep the target variable unaltered.

We trained the model over variable folds, keeping the learning rate 0.0001 and the number of epochs is kept at 5000. The model performed well at 80%, 20% train-test split. We employed cross-validation for splitting the dataset. As seen in table 5, logistic regression attained the highest f-score at 9:1 train-test data split. While training the logistic regression model on 2 variables (Risk and Flu), there was no false negative detected. Hence the recall of the model remains 1.0 throughout the experiment. The confusion matrix of the experiment is provided in table 6.

Since the domain and range of the independent variable and dependent variable is different, we

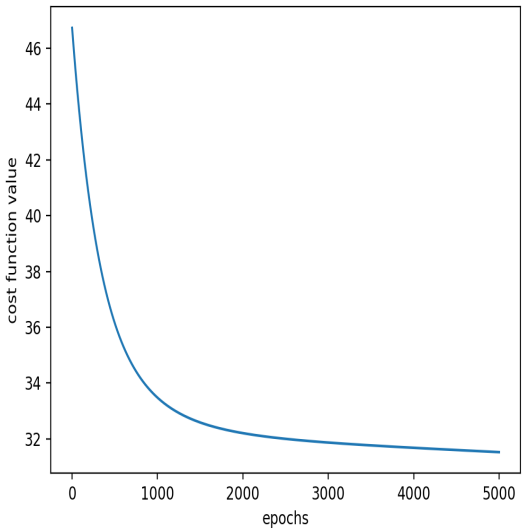


Figure 5: Training error of Logistic regression model with one variable

applied feature scaling (normalization).

$$\text{scaled-dataset} = \frac{feature^i_j - \mu_{feature^i}}{max(feature^i) - min(feature^i)}$$

i ϵ {*KnowlTrans*, *Risk HndWshQual*} and j ϵ [0, length($feature^i$)-1]

We applied Min-Max normalization for our experiment Confusion matrix shown in table 6, is for the experiment in which the logistic regression achieved maximum f-score. This matrix helps in the identifying the number of true-positives, false-positives, true-negatives and false-negatives. We read the table 6 as; the model captured 12 true values out of 12 true values present in the test set. Since the model has one features, there was less probability of generating the true-negatives. In the subsequent sections, we will keep on augmenting a variable to the list of independent variables with dependent variable being *Flu*.

4 Logistic Regression with Multiple Variables

In this section, we will train the logistic regression model on a multi-variable dataset, which means there are more than one independent variable but one dependent variable. Furthermore, we will assess augmenting which features significantly improved the model performance. In this experiment and also the previous experiment, we haven't used regularization of the cost function. We are utilizing stochastic gradient descent as the optimization function for the learning of the model.

4.1 Importance Risk and HndWshQual

Yes, we can map Risk and HndWshQual to Flu. The domain of Risk and HndWshQual is stated in table 1. Now, we have to see what impact does HndWshQual has on the logistic regression model. Seeing the f-score in table 7, we see the model trained over this dataset showed an

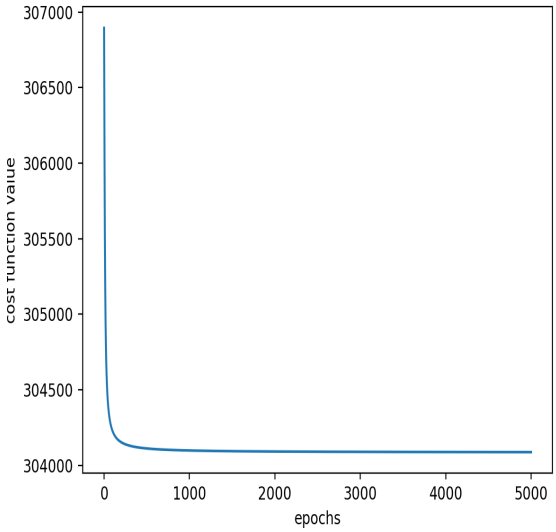


Figure 6: Training error of logistic regression model with two variable

Table 7: Cross Validation Fscore results tables. LogR2V : Logistic Regression with two Variables

	%age Training	%age Testing	Precision	Recall	F-score
	20	80	0.26	0.78	0.39
	30	70	0.33	0.82	0.47
	40	60	0.30	0.83	0.44
	50	50	0.39	0.60	0.47
	60	40	0.34	0.86	0.489
	70	30	0.32	0.60	0.42
	80	20	0.39	0.50	0.44
	90	10	0.32	1.0	0.486

averaged f-score of 0.45 which is 13% improvement over the model trained in table 5. Hence, adding *HndWshQual* feature to the model improved the model. This improvement came at the cost of model detecting 2 true-negatives (table 8) which is compensated by the decline in the number of false positives, in comparison with table 6.

We observe the training of the model by plotting the cost function value with the number of epochs. Henceforth, we compare the training curve with the addition of more features of the dataset. On comparing with figure 5, adding a new variable made the curve more steep and flattened, observable from figure 6.

Addition of *HndWshQual* to the dataset, improved the model efficiency. As evident from table 7, there is a non-monotonous increase in the f-score with increase in the training size with an outlier result lying at 9:1 train-test split. It is quite trivial for model to have high recall at 9:1 train-test split. This is because, the model learned patterns from large number of observations in the training data and possibly have overfitted. In order to regulate the learning of the model, one can either use large number of features for the model training or perform regularization of

Table 8: Confusion matrix of the logistic regression with two variable. Threshold kept very low : 0.2

	Predicted-True	Predicted-False
Actual-True	12	2
Actual-False	23	25

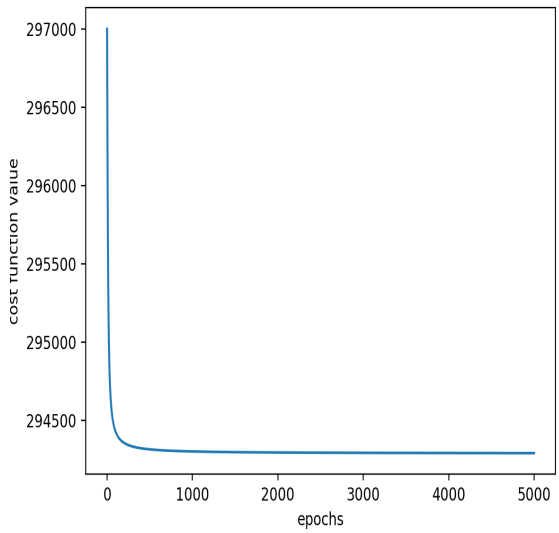


Figure 7: Training error of logistic regression model with three variables

the cost function so that model learns and not generalize.
The confusion matrix is something similar to contingency table but it is more of assessing the quality of prediction made by the model. On observing, logistic regression with two variables confusion matrix, we see improvement in terms of reduction in the number of false positives.

4.2 Does KnowlTrans improve the performance of the model?

Yes, the addition of *KnowlTrans* to the dataset, improves the performance of the logistic regression model. Average f-score increased from 0.450 to 0.453. A 0.4% improvement in the model efficiency is attributed to *KnowlTrans* because of the reduction in the variance in the dataset. *HndWshQual* is a Likert scale feature which is scaled between 0 and 1, resulting in the addition of variance in the dataset which is balanced by *KnowlTrans*. There is more steepness in the training error, which is evident from figure 7. Such asymptotic learning curve of the model states the model has been trained with tuned parameters and probability of overfitting is very less. Now the question is can we do better with regularization.

In comparison with table 7, logistic regression model trained over the dataset with three variables (KnowlTrans, HndWshQual, Risk) showed improvement in precision values. Table 9 shows an improvement of 10% in average precision. This shows that addition of *KnowlTrans* improved model efficiency. Also, we observe improvement in the confusion matrix. A positive

Table 9: Cross Validation Fscore results tables. LogR3V : Logistic Regression with three Variables

	%age Training	%age Testing	Precision	Recall	F-score
	20	80	0.33	0.36	0.35
	30	70	0.30	0.66	0.41
	40	60	0.33	0.75	0.45
	50	50	0.34	0.61	0.44
	60	40	0.38	0.73	0.50
	70	30	0.40	0.85	0.54
	80	20	0.42	0.54	0.47
	90	10	0.43	0.50	0.46

Table 10: Confusion matrix of the logistic regression with three variables. Threshold: 0.2

	Predicted-True	Predicted-False
Actual-True	11	2
Actual-False	17	23

insight using the confusion matrix (table 10) is that there is a decrease in the number of false positives by the model. In all our comparisons, we have generated the confusion matrix keeping the threshold for classification equals to 0.2.

4.3 Assessing model improvement after adding Gender as feature

Gender is a binary variable in the flu dataset. Because of its binary nature, we did not perform normalization (feature scaling) over this feature. Hence the features scaled in this experiment remain the same as previous experiments (KnowlTrans, HndWshQual, and Risk). In our experiments, an addition of Gender features did not improve the model performance. Based on the cross-validation results in table 11, averaged f-score of the model is 0.443 and averaged precision is 0.36. In comparison with table 9, there is a reduction of 2% in the f-score and 0.7% in precision. Though the reduction is minuscule when we work on the big-data problem where observation size is in tens of millions, these reductions are amplified.

In spite of a reduction in f-score and precision, this logistic regression model attained a precision of 50% in this noisy dataset, which is by far the highest in comparison to our previous experiments. The trends in f-score did not remain same in this experiment. In this experiment, 9:1 train-test split performed well over other data split ratios (similar to LogR1V but did not overfit). The reason for this behavior is attributed to a large amount of information is available for the model in terms of Gender, KnowlTrans, HndWsh, and Risk.

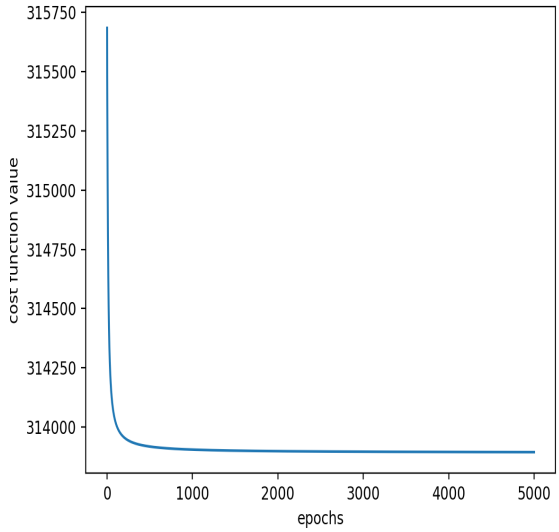


Figure 8: Training error of logistic regression model with four variables

Table 11: Cross Validation Fscore results tables. LogR4V : Logistic Regression with four Variables

	%age Training	%age Testing	Precision	Recall	F-score
	20	80	0.27	0.41	0.32
	30	70	0.29	0.55	0.38
	40	60	0.32	0.71	0.44
	50	50	0.50	0.45	0.48
	60	40	0.38	0.66	0.49
	70	30	0.33	0.78	0.47
	80	20	0.32	0.67	0.43
	90	10	0.45	0.63	0.53

Table 12: Confusion matrix of the logistic regression with four variables. Threshold: 0.2

	Predicted-True	Predicted-False
Actual-True	5	3
Actual-False	6	25

We compare the efficiency of the model in this experiment with the logistic regression model created with one variable. Our reason for bringing out this comparison is the fact that in both the scenario, the model performed well on 9:1 train-test split. It is conclusive from table 12, that there is a tremendous decrease in the number of false positives. As seen from figure 1, the number of 0.0 values is large as compared to 1.0 under *Flu* label. This is correctly portrayed by the confusion matrix in table 12. Though the addition of more features resulted in an increase of False-negatives it is very well compensated by the increase in true-negatives. Overall, this model performed fairly well in comparison with one variable and two variables models.

5 Impact of Regularization and Feature Scaling

In this section, we experiment with the regularization component of the cost function and the regularization parameter (λ). Before, detailing more about the regularization and $\lambda's$, we define regularization.

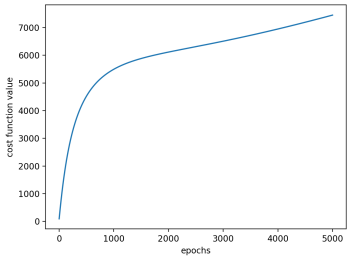
Regularization is a technique used to prevent the model from over-fitting by adding a high weighted term to the optimization function. There are various ways of formulating this terms, such as , L1 ($||\theta||$), L2 ($||\theta||_2^2$). We have used Frobenius norm [] to formulate our optimization function with regularization.

$$J(\theta) = \sum_{i=1}^{N_{tr}} (h(\theta)_i - Y_i)^2 + \lambda ||\theta||_2^2$$

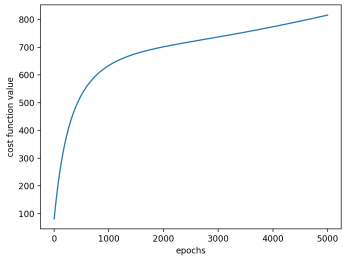
where $J(\theta)$ is the cost function, θ are the weights,
N_{tr} : Number of the training samples. This value changes with the cross folds ratios.
λ: this is the regularization parameter.
$h(\theta)_i$: this is the hypothesis function.
$h(\theta)_i = \frac{1}{1+e^{\theta^T x_i}}$; x_i is i^{th} observation in the dataset. Y_i : is the i^{th} label under *Flu*.

Regularization and feature scaling are the techniques used to make model reach its goal of convergence and better prediction without over-fitting and preventing the ill-posed problem. I have selected logistic regression with three variable for regularization. The reason of our selection is stated in details in the previous section. A quick recap, *Gender* feature did not add improvement to the model. So, after regularized learning of logistic regression model, we obtained an averaged f-score of 45% and averaged precision of 37%. This states that there is a decline of 0.6% in the f-score and an increase of 2% in precision, compared with logistic regression model with three variables. This is a considerable improvement over previous experiments.

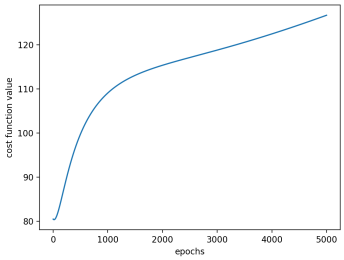
The regularized logistic regression model performed well at 8:2 train-test split of the dataset. Comparing the table 9 and 13, we see that there is an improvement in precision and f-score at 8:2 train-test split. 6:4 and 7:3 train-test split in table 9, which gave the highest f-score (top-2), gave high recall whereas the regularized model gave low recall and at par (also higher) comparable precision in both the train-test splits (decrease in recall is 1.6%). This certainly explains the need of regularization in the model training, especially if the data is noisy.



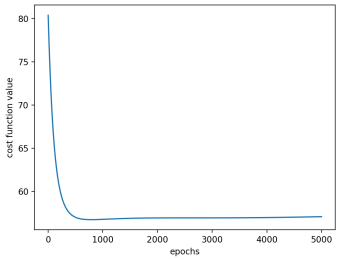
(a) lambda = 20



(b) lambda = 2



(c) lambda = 0.2



(d) lambda = 0.02

Figure 9: Training error curve when the cost function is regularized with (a). lambda = 20 (b). lambda = 2 (c). lambda = 0.2 (d). lambda = 0.02. Lambda is a paramter of the cost function that involves a regularization component where the lambda parameter is multiplied by the frobenius norm of the weights ($\theta's$ or coefficients) of the features in the dataset.

Table 13: Cross Validation Fscore results tables. RegLogR3V : Regularized Logistic Regression with three Variables

	%age Training	%age Testing	Precision	Recall	F-score
	20	80	0.28	0.68	0.39
	30	70	0.29	0.63	0.40
	40	60	0.33	0.73	0.46
	50	50	0.32	0.71	0.44
	60	40	0.40	0.53	0.46
	70	30	0.38	0.64	0.47
	80	20	0.50	0.50	0.50
	90	10	0.42	0.50	0.45

Table 14: Confusion matrix of the Regularized logistic regression with four variable. Threshold: 0.2

	Predicted-True	Predicted-False
Actual-True	7	7
Actual-False	8	24

The confusion matrix generated in this model is associated with 8:2 train-test split results. Observing the table 14, regularized model gave the specificity score of 75% which is by far greater than 57.5% specificity score of un-regularized logistic regression (LogR3V).

Note : Specificity is defined as the ratio of true-negatives over the sum of true-negatives and false-negatives. Apart from regularization ,feature scaling is another technique for improving the performance of the models. Feature scaling (in our case : Min-Max normalization) or normalization is a process of the bringing all the feature values within the same range. In this assignment we scaled all the feature values between [0,1]. This is essential because, unscaled feature values abnormally alter the weights of the features and disturbs the convergence and classification efficiency of the model. For instance; a feature with values lying in the range [1000, 6000] will have more weight (importance) than feature with values lying in the range [0.1, 0.6]. Hence, feature scaling is important. In our dataset, Flu and Gender features have values in the same range as stated in table 1 whereas KnowlTrans, HndWshQual and Risk have different range (table 1). Hence we need to scale values of KnowlTrans, HndWshQual and Risk to [0,1]. We applied Min-max scaling approach in our assessment. Since, the feature values were not scaled and lack of normalization, affects the pattern recognition capability of the model.