ES7023 - Assignment 6: Model Evaluatione

Model evalutation is one of the key steps to the statistical model development process, and is necesary before you start using your model for real prediction. In this assignment we will focus on a classification problem for landslide prediction. You have seen this data before in assignment 3 and in class, but now we are using a larger dataset. You are provided with three data files: (1) a training set (n=300), (2) validation set (n=100), and (3) a test set (n=100).

The data consists of the following parameters:

- `x` & `y` : coordinates
- `lslpts` : True (landslide occured), False (no landslide). This is the variable we will try to model and predict.
- `slope`: slope angle (°).
- `cplan`: plan curvature (rad m^{-1}) expressing the convergence or divergence of a slope and this water flow.
- `cprof`: profile curvature (rad m^{-1}) as a measure of flow acceleration, also know as downslope chane in slope angle.
- `elev`: elevation (m above sea level) as the representation of different altitudinal zones of vegetation and precipitation in the study area.
- `log10_carea`: the decadic logarithm of the catchment area (log10 m^2) representing the amount of water flowing towards a location.

Read in the following data files:

```
#Set to the correct directory where you have placed your files
setwd("/Users/david/Dropbox/My Mac (ASE-Mac2141)/Documents/R programing/")
#Read the training data-set
landslide.train=read.csv("landslide_training_data.csv",header = T)
#Read the validation data-set
landslide.validation=read.csv("landslide_validation_data.csv",header = T)
#Read the test data-set
landslide.test=read.csv("landslide_test_data.csv",header = T)
```

Problem 1 - Data Exploration and tidying

Note: For this problem you will use only the training data: `landslide.train`.

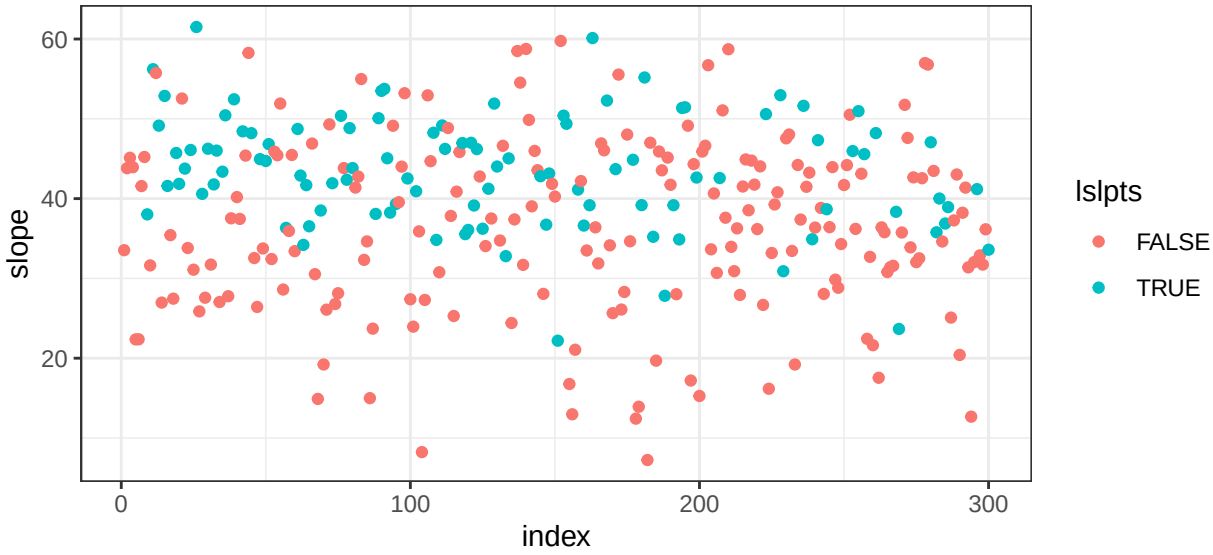
1.a Data Exploration

- Develop histograms of the five different predictor variables in the training set. Comment on the parameter ranges and distributions, and potential outliers.
- Develop a correlation plot of the predictor variables.

1.b. Outliers

Treating or removing outliers / extremes for genuine observations is typically not recommended. In fact sometimes the extremes are specifically the observations of interest (example: flood analysis requires study of extreme river discharge). Sometimes however, outliers can indicate errors in measurements or other glitch in our data-collection or processing. In the context of modelling and prediction, outliers can sometimes drastically impact the fit / bias of our model. Remember for instance, that loss functions related to squared errors are highly sensitive to outliers.

- Plot each of your predictor variables vs its index (row number), and assign point color corresponding to landslide or no-landslide. *Note:* you might have to add a column to your data frame with the index of each data point. It should look something this:



- In each plot, add horizontal lines corresponding to $\mu \pm 2\sigma$ (mean value of the variable $\pm$ 2 standard deviations) and $\mu \pm 4\sigma$.
- Comment on outliers with reference to your plots.
- Look up discussions on "treating outliers" online. Explain in 1 sentence each the terms "imputation" and "capping" in the context of outliers.
- Briefly discuss a proposed course of action for treating outliers in the data.

For now let's just keep our data as is, outliers included, and move on to normalizing.

1.c. Normalization and Standardization

Sometimes it is good to normalize variables. Many optimization algorythms for instance, are unstable when multiple parameters are used with vastly different scales. This is why we normalized the elevation data for you in assignment 3, as it otherwise caused instability in your self-built MLE maximizer. Standardizarion helps with convergence of many optimization algorithms. In addition, standardization also leads to a different interpretation of the coefficients (now called *standardized coefficients*), since all predictor variables are now all on the same scale.

There are many ways to standardize/normalize data. 'Z score standardization' is one of the most popular. It rescales and centers the data so that it has mean value of zero and standard deviation of one. The standardization is performed using the following equation:

$$z = \frac{x - \mu_x}{\sigma_x} \tag{1}$$

where μ_x and σ_x are the mean and standard deviation of x respectively (x is a predictor variable).

- Develop a new data frame `landslide.train.standard` with standardize predictor variables using the Z score equation above.
- Develop histograms to discuss how your data has been transformed.
- Develop a correlation plot of the predictor variables. How does this compare with the correlation plot of the un-normalized variables. Is this what you expected? Why?

Problem 2 - Model Development - Logistic Prediction

Now lets build our landslide prediction model! We will use R's built-in generalized linear model function `glm()`. Refer to assignment 3 for description of generalized linear models. Below is a simple example demonstrating how to use the `glm()` function with some simulated data:

```
#Creating a data-frame of simulated data
my.data=data.frame(response=c(rep(0,20),rep(1,20)), var1=c(rnorm(20,mean = 5,sd = 5),+
 rnorm(n = 20,mean=10,sd=5)),var2=c(rnorm(20,mean = 2,sd = 5), rnorm(n = 20,mean=4,sd=5)))
#Fit a logistic model
my.model=glm(formula = response ~ var1+var2,family = binomial('logit'), data = my.data)
summary(my.model)
```

2.a. Original data

- Use the original training data `landslide.train` to fit a logistic model for predicting landslides. Show the summary of the model (as in the example above).
- Discuss the interpretation of the model coefficients.
- Compute the Root-Mean-Squared-Error of the fitted model. *Note* that among the various outputs of the `glm()` function are the residuals. In the simulated example above, residuals are obtained using `my.model$residuals`.

2.b. Standardized data

- Use the standardized training data `landslide.train.standard` to fit a new logistic model. Show the summary of the model.
- Discuss the interpretaion of the coefficients.
- Compute the Root-Mean-Squared-Error of the fitted model. How does it compare to that computed for the model fit with the original data? Is this what you expected?

Note: when using a normalized data-set in modeling, one needs to be careful with the validation and test data-sets. If one includes the validation and/or test data in the normalization, then the validation/test sets are 'leaking' information into the model training (i.e. the validation and test set influence the training of the model). So the correct way to do this is to standardize only based on the training data, such that:

$$z_{training} = \frac{x_{training} - \mu_{xtraining}}{\sigma_{xtraining}} \tag{2}$$

$$z_{validation} = \frac{x_{validation} - \mu_{xtraining}}{\sigma_{xtraining}} \tag{3}$$

$$z_{test} = \frac{x_{test} - \mu_{xtraining}}{\sigma_{xtraining}} \tag{4}$$

We will continue our study using the original model fit with the non-normalized variables.

Problem 3 - Model Prediction

You will use the logistic model you developed in Problem 2 (the one fit to non-normalized data) to predict outcomes on the `landslide.validation` data. Once you have calibrated the `glm` model on the training data, R has some convenient functions to predict outcomes of your model based on new data inputs. Using the simulated example described previously, predicting response is done using the following code:

```
#create a data-frame for new data
prediction.data=data.frame(var1=c(6,3,11,12,3),var2=c(1.3,2,4,4.5,2.1))
#predict the response of your model with new data
pred=predict(object = my.model, newdata=prediction.data, type="response")
```

- Predict the logistic response on the validation data-set `landslide.validation`. The response should be a value ranging from 0 to 1, corresponding to the odds of landslide.

- Develop a confusion table for landslide occurance, using a threshold of 0.5.
- Calculate the *accuracy* of your model.

Problem 4 - Model Selection

Whenever possible, we should trial several alternative models. These can be models with varying functional forms, varying parameters, or models that are conceptually completely different. You are invited to create an alternative model of your choice to predict landslides. Here are some ideas:

- *Logistic model* with one or several transformed predictor variables
- *Logistic model* with outliers removed or capped
- *Logistic model* with fewer predictor variables (remember that sometimes the model might be overfitting the training data)
- *Logistic model* with interaction term(s) (you can find an example here: https://www.theanalysisfactor.com/generalized-linear-models-glm-r-part4/)
- *Probit model.* In general logistic and probit models tend to have very similar outputs, but look up "probit vs logit models" to learn about the difference.
- *Complementary Log-Log model.* Another generalized linear model. *Note:* logit, probit and complementary log-log models are all generalized linear models, available in R's built-in `glm()` function. Look up the function for details (type `?glm()` in the R-Studio console), and read this article for summary differences between the models: https://data.princeton.edu/wws509/notes/c3s7

Many more potential models. Only explore these if you are particularily interested and have the time, as these are beyond the scope of the class.

- *k-nearest neighbors model.* This is a non-parametric (no coefficients) data-driven algorithm.
- *Generalized Additive Models (GAM)* See Hastie and Tibshirani (any of their books since 1990)

Once you have chosen a new model to test:

- Fit your new model to the training data set.
- Summarise the model fit and interpretation (e.g. coefficients).
- Use your new model to predict the response on the `landslide.validation` data-set.
- Create a confusion table of the results, and calculate the accuracy.
- How does your 2nd model perform compared to the logistic model from problem 2 and 3?
- Based on the results (or other error metric of your choice) select a single model and explain your selection.

Problem 5 - Model Evaluation

There are many different metrics to evaluate model performance. In our problem, the response is a binary variable describing the occurence or non-occurence of a landslide. The model we are developing is therefore essentially a *classification model*. We will explore several standard metrics for evaluating classifications models. For this problem, you will be using your chosen model from Problem 4, and the `landslide.test` data-set.

5.a. Model skill metrics

Compute the following model skills metrics:

- Model accuracy
- Model precision
- Model recall
- Model F1-score

5.b. Brier Score

The Brier Score is useful for assessing model performance when predictions are probabilistic. We can therefore use the predicted response of your model (the predicted response is a value between zero and one) to calculate the Brier Score. *Note* you can calculate the Brier score for the generalized linear model or generalized additive

models. If you selected a k-nearest neighbor model or other non-probabilistic classification model, you will not be able to calculate the Brier score. In that case calculate it for the logistic model you developed in problem 2 and 3.

The equation for the Brier Score is the following:

$$BS = \frac{1}{n}\sum_{i=1}^{n}(fp_i - v_i)^2 \tag{5}$$

where fp_i is the predicted response probability and v_i is the observed outcome.

- Calculate the Brier Score for your model.
- Is the Brier Score a good measure of model skill for our problem? Explain your reasoning.

5.c. ROC and AUC

The measures you calculated in problem 5.a. are dependent not only on the overall model performance, but also on the choice of *classification threshold* (also called *decision threshold*). The ROC curve (receiver operating characteristic curve) is a plot showing the performance of the classification model at all classification thresholds. It is the plot of *True Positive Rate (TPR)* vs *False Positive Rate (FPR)* for all thresholds. Based on the ROC curve, we can compute the *Area Under the ROC Curve (AUC)*. It is a single value which provides an aggregate measure of the performance of the classification model accross all possible classification threshold.

- Compute the True Positive Rate and False Positive Rate sequetially for all thresholds between 0 and 1 with increments of 0.01.
- Plot the ROC curve
- Compute the AUC
- Comment on the model performance

5.d. ROC and AUC for your alternative model (OPTIONAL) This is an optional problem, if you want to explore this further.

- Compute the ROC and AUC for an alternative model developed in Problem 4
- Plot on the same plot as in 5.c.
- Comment on the results