CeADAR – Centre for Applied Data Analytics Research

Enterprise Ireland Data Analytics Technology Centre

Regression Model Visualisation - State of the Art Review

Document Type:	Literature Review
Project Title:	CeADAR
DDN Theme:	1 – Intelligent Analytic Interfaces
Theme Leader:	Sarah Jane Delany (DIT)
DDN Sub-Theme:	Ease of Interaction
Authors:	John Kelleher
Document Version:	0.2
Date of Delivery to ISG:	2nd March 2015
Number of pages:	9

ABSTRACT

In this review, we discuss the state-of-the-art of both research and development in visualisation techniques that: (a) can be used to visually communicate the uncertainty associated with predictions from a regression model, and (b) allow non-experts to understand the reasoning behind the output from regression prediction models.

CeADAR is a research partnership comprising University College Dublin, University College Cork, and Dublin Institute of Technology.

http://www.ceadar.ie

Contents

1 Introduction

This document is organised in three main sections. First, we describe what a regression model is. Second, we describe some of the standard visualisations used to: (a) analyse the behavior of regression models; and, (b) explain regression models to businesses. In the third, and final part, of the document we present our conclusions and recommendations for the project.

2 What is a Regression Model

One of the most popular types of prediction model is a regression model. The simplest type of regression model is a linear regression model, which has the following form:

$$\mathcal{M}_w(\mathbf{d}) = w[0] \times \mathbf{d}[0] + w[1] \times \mathbf{d}[1] + \cdots + w[m] \times \mathbf{d}[m] \quad (1)$$
$$= \mathbf{w} \cdot \mathbf{d} \quad (2)$$

where $\mathcal{M}_w(\mathbf{d})$ is the prediction made by the model for a given vector of input features $\mathbf{d}$, $w[i]$ is a weight learned by the model for each input feature, and $\mathbf{d}[i]$ describes the value of the i^{th} input feature. Note, the weight $w[0]$ is slightly different from the other weights; this weight defines the x intercept of the model and the input feature for this weight ($\mathbf{d}[0]$) is always set to 1.

For example, imagine we have trained a regression model to assess the credit risk of a loan that works with just two features:

1. LOANSAL, the ratio of the loan amount to the applicant's salary
2. DEBTSAL, the ratio of the applicant's current debt to the applicant's salary
3. COLLLOAN, the ratio of the applicant's collateral to the loan amount

$$\mathcal{M}_w(\mathbf{d}) = 6.2 + 0.07 \times \text{LOANSAL} + 0.9 \times \text{DEBTSAL} + -0.5 \times \text{COLLLOAN} \quad (3)$$

Analyzing the weight in this models tells us that a unit increase in the LOANSAL increases the risk prediction of the model by 0.07 whereas a unit increase in the DEBTSAL input feature increases the risk prediction of the model by 0.9, and each unit increase in the ratio of collateral to loan amount decreases the risk by 0.5. This types of analysis of a model is often important to a business.

The model defined in Equation 2 is suitable for making predictions for continuous numeric features. If we wish to use a regression model to make a categorical prediction the standard approach is to push the output of the regression model through a logistic function (this forces the predictions into the range $[0 \ldots 1]$) and then apply a threshold to the result. This type of model is known as a logistic regression model and is defined as

$$\mathcal{M}_w(\mathbf{d}) = \begin{cases} Class1 & \text{if } \frac{1}{1+e^{(\mathbf{w}\cdot\mathbf{d})}} > Threshold \\ Class2 & \text{if } \frac{1}{1+e^{(\mathbf{w}\cdot\mathbf{d})}} \leq Threshold \end{cases} \quad (4)$$

Mapping this logistic regression model to the credit risk domain we introduced earlier and assuming a threshold value of 0.5 the example model would have the following structure:

$$\mathcal{M}_w(\mathbf{d}) = \begin{cases} \text{High Risk} & \text{if} \quad \dfrac{1}{1 + e^{(6.2 + 0.07 \times \textsc{LoanSal} + 0.9 \times \textsc{DebtSal} + -0.5 \times \textsc{CollLoan})}} > 0.5 \\ \text{Low Risk} & \text{if} \quad \dfrac{1}{1 + e^{(6.2 + 0.07 \times \textsc{LoanSal} + 0.9 \times \textsc{DebtSal} + -0.5 \times \textsc{CollLoan})}} \leq 0.5 \end{cases} \quad (5)$$

A nice feature of logistic regression models is that the prediction of the model prior to thresholding can be interpreted as the probability of the predicted class. These probabilities are also often useful to businesses.

3 Regression Model Visualisation

There are two distinct goals that a visualisation of a fitted regression model can have:

1. the visualisation can be designed as a diagnostic aid that provides the data analyst with insight into the behavior of the fitted model with respect to the impact of changing the value of an input feature, or the interaction between different features in a model, etc.
2. the visualisation can be designed to assist with explaining to a business user what a regression model is doing.

In Section 3.1 we will overview the standard visualisations used as a diagnostic aid to examine a model structure. Following this, in Section 3.2, we will describe a standard way of presenting a regression model to a business user.

3.1 Model Diagnosis

Typically, with regression analysis, we wish to understand the uncertainty around a prediction made by a model for a specific vector of descriptive features. Including **confidence intervals (or bounds)** based on standard errors (e.g. Wald confidence intervals) within a visualisation is often done to illustrate this uncertainty. Note that a regression line between two variables, an input variable X and a predicted variable Y, plots the estimated mean of the population of the predicted variable at any value of input variable (see Fig 1). A confidence interval defines a range of values with a specified probability that the value of a parameter lies within it. Given this, it can be useful to plot the uncertainty with respect to the prediction of a regression model by plotting the confidence interval around the regression line for the estimated mean parameter.

Plotting a regression line with confidence intervals is the standard visualisation for regression models. However, in terms of model diagnosis there are a number of other aspects to a model that we may wish to examine. For example:

1. We may wish to understand the relationship between an independent descriptive feature and the binary target feature after fixing the values of one or more of the other descriptive features in the domain. Here visualisations such as **conditional plots** and **contrastive plots** are useful. A **conditional plot** shows the values of a particular descriptive feature on the x-axis and the change in the probability of the positive target

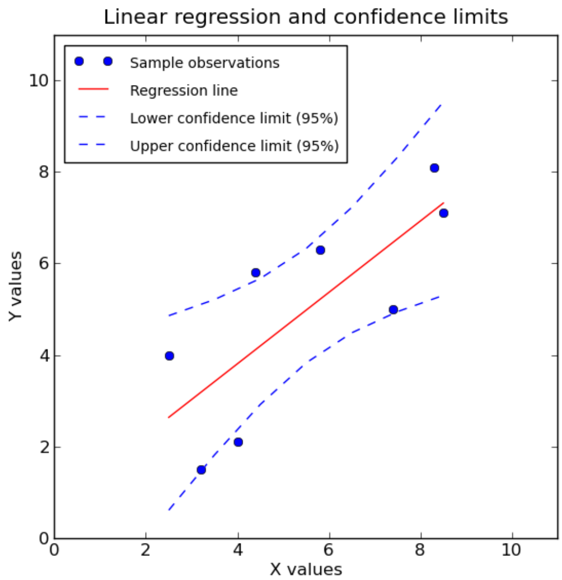

Figure 1: A plot of a regression line with the confidence bound also plotted. Image retrieved from [3].

outcome on the y-axis, holding all the other descriptive features constant (by default, the median for continuous features and most common level for categorical features). A **contrast plot** has a similar structure to a conditional plot; the difference being that a **contrast plot** shows the effect on the expected probability of the positive target outcome as we move the value of one of the descriptive features away from some reference point on the x-axis while holding the other descriptive features constant (for continuous features this reference points is taken to be the mean). Figure 2 illustrates the conditional plots generated for each of the input features in the linear regression model listed in Equation 6.[1] This model was trained on a standard R dataset dealing with air quality prediction. Figure 3 illustrates the contrastive plots generated for each of the input features in the same model.

$$Ozone = Solar.R + Wind + Temp \tag{6}$$

2. We may wish to understand the effect of varying the value of a particular descriptive feature on the target feature outcome. A standard way of doing this analysis is to examine the regression coefficient of the descriptive feature. However, if there are **interactions** between the descriptive features in the domain then this approach breaks down. In these situations visualisations, such as **cross-sectional plots** are useful. A **cross-sectional plot** illustrates the one-dimensional relationship between the target variable and one of the descriptive features for several values of another descriptive feature. Figure 4 shows a set of cross sectional plots showing the relationship between the target feature (Ozone) and one of the descriptive features (Wind) for different levels of a categorical descriptive feature (Heat), which has three levels (Cool, Mild, and Hot) and which has an interaction effect on the descriptive feature in the plots. The model used for these plots is defined in Equation 7 and was trained on the same air quality dataset

[1] This example, like the next example and visualisations in this document, is based on examples in [1].

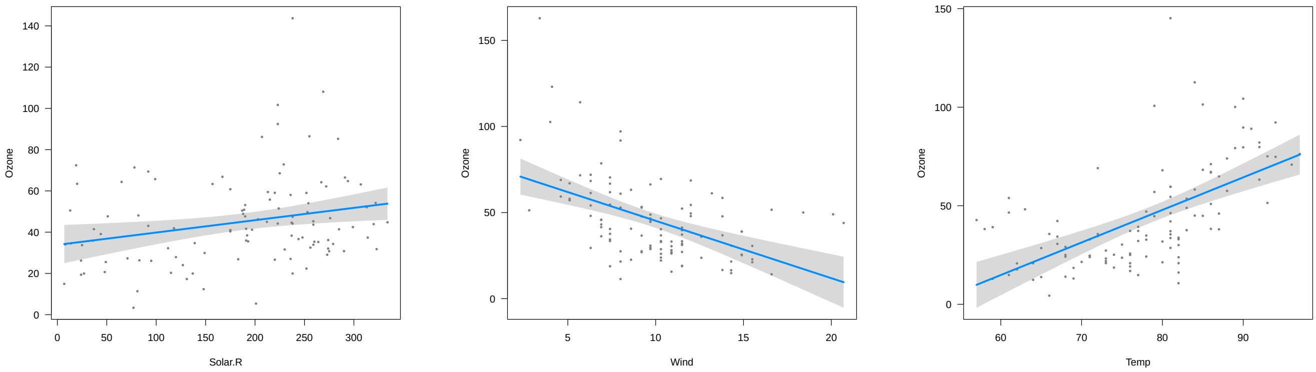

Figure 2: Conditional plots for each of the descriptive features in Equation 6 once the model had been trained on the air quality dataset.

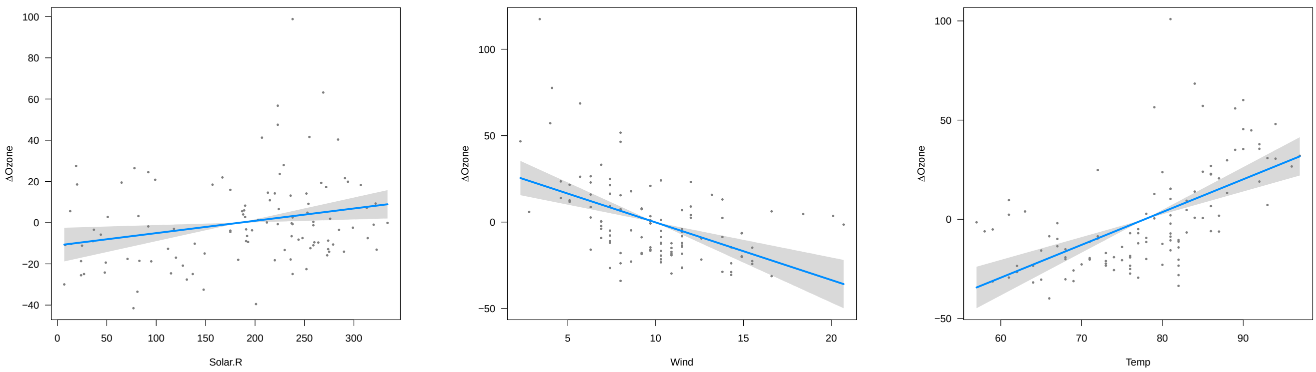

Figure 3: Contrastive plots for each of the descriptive features in Equation 6 once the model had been trained on the air quality dataset.

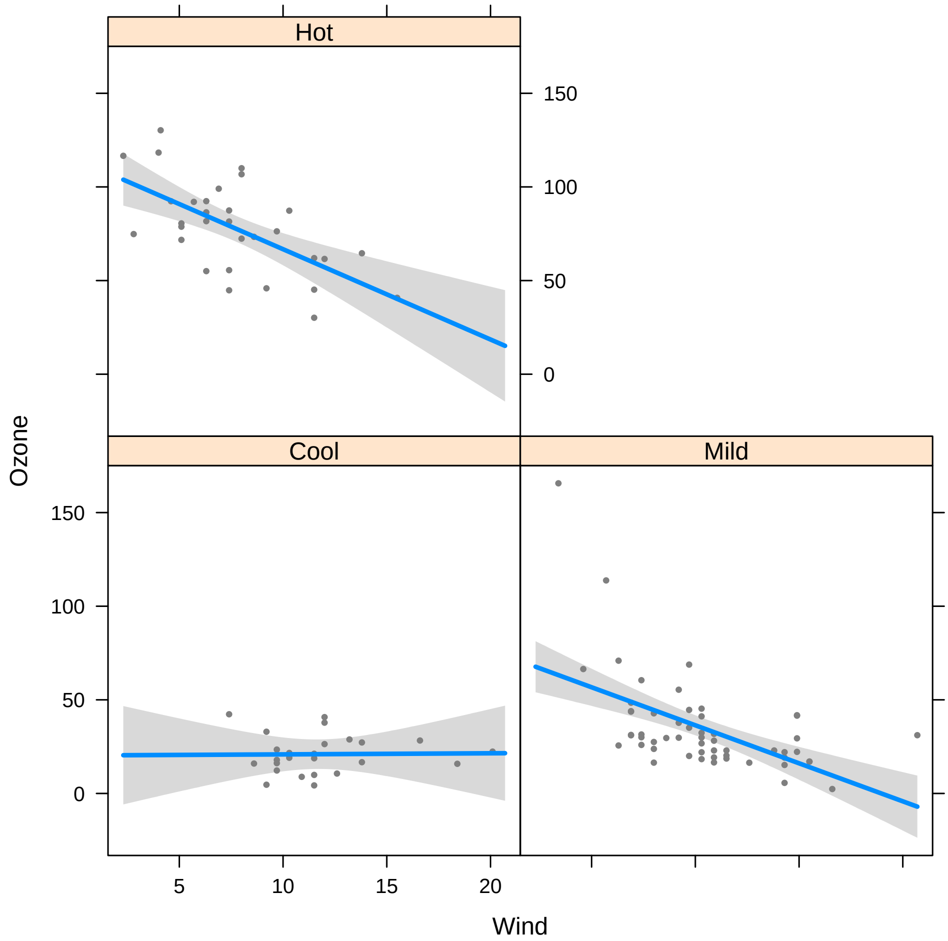

Figure 4: Cross-sectional plots showing the relationship between the target Ozone feature and the descriptive feature Wind for different levels of the Heat descriptive feature which has an interaction with the Wind feature.

as used above.

$$Ozone = Solar.R + Wind \times Heat \quad (7)$$

3. We may wish to understand what a particular descriptive feature adds to the model over and above the other descriptive features in a model. **Added-variable plots** are a standard visualisation used to illustrate this type of information. An **added-variable plot** plots the **residuals** for the logistic regression model built using the subset of descriptive features that excludes the descriptive feature contribution we are analysing versus the residuals of a logistic regression model on predicting the descriptive feature we are analysing using the same subset of descriptive features as input to the model. One benefit of added-variable plots is that they enable the analyst to ascertain whether the contribution made by a descriptive feature to a model is sustained across the range of the descriptive feature, or if it is only due to a few (influential) data points. We won't provide an example of an added-variable plot here as it would require a relatively long explanation.[2]

[2]See www.academia.edu/2519488/WHY_ADDED_VARIABLE_PLOTS_ for more on added variable plots.

Table 1: Example scorecard

Feature	Score
LoanSal-High	20
LoanSal-Low	5
DebtSal-High	40
DebtSal-Low	10
CollLoan-High	5
CollLoan-Low	40
Total score	

There are an array of visualisation solutions for regression models ranging from freely available libraries to functionality embedded within the large commercial analytics offerings. An example of a freely available library is the R package **visreg** [2]. This package provides contrast, conditional, surface and cross-sectional plots. It is also includes confidence bounds within plots, where appropriate. Most of the standard commercial analytics packages (e.g. SPSS, SAS and STRATA) include visualisation packages for regression models. [2]

3.2 Model Explanation

In contrast with the visualisations designed to help a data analyst diagnose the behaviour of a regression model, there are very few methods for explaining a regression model to a typical business user. In fact, often regression models are presented to businesses in the form of a **scorecard**.

A scorecard is composed of a set of features each of which is assigned a score. For example, Table 1 illustrates a potential scorecard representation for the logistic regression model in Equation 5. Using this scorecard a loan application with a high score would be classified as high risk and a loan application with a low score would be classified as low risk.

Transforming a regression model into a scorecard is non-trivial.[3] For example, to construct the scorecard in Table 1 each of the continuous features in the model in Equation 5 were converted into a binary categorical feature using binning and then represented as two binary features (one feature for each level of the binary categorical feature). A regression model was then trained using the binary features and the weights in the resulting model were then scaled to create scores in the desired range.

The advantage of a scorecard representation of a regression model is that they are relatively easy to understand. However, as noted above, designing and building a scorecard can be difficult.

4 Conclusions

There are a lot of visualisations that can be used by data analysts to examine and diagnose the behavior of a regression mode. Furthermore, there are also a number of different soft-

[3] [4] provides an excellent introduction to the design and implementation of scorecards based on regression models in the credit risk domains.

ware solutions that offer APIs to create these visualisations. By contrast, there is very little work done on developing visualisations that explain to businesses what a regression model is doing. Although scorecards are often used for this purpose they can be difficult to design and build. Given this, we believe that a useful target for this project would be to develop visualisations of regression models that provide an intuitive understanding of how the model is making prediction decisions in terms of the weighting and contribution of each feature to the overall outcome.

References

[1] Patrick Breheny and Woodrow Burchett. *Visualization of Regression Models Using visreg*. http://myweb.uiowa.edu/pbreheny/publications/visreg.pdf, May 2013.

[2] Patrick Breheny and Burchett Woodrow. *Package 'visreg': Visualization of Regression Models*. http://cran.r-project.org/web/packages/visreg/visreg.pdf, 2015.

[3] Tom Holderness. https://tomholderness.files.wordpress.com/2013/01/linearfit.png.

[4] Naeem Siddiqi. *Credit Risk Scorecards: Developing and Implementing Intelligence Credit Scoring*. Wiley, 2006.