Latent Dimensionality Reduction: An Algorithmic Method for Interpreting Black Box Models

DRAFT

Elias Kassell[1] and Fred Farrell[2]

[1]eliaskassell@gmail.com
[2]ffarrell@illumina.com

September 2019

1 Abstract

LDR (Latent Dimensionality Reduction) is an algorithmic method for interpreting models by decomposing predictions from the original high dimensional feature space into a lower dimensional space. Both the output of the model and the certainty of its predictions are encoded in the condensed space. The method works out of the box on both classification and regression problems with input features relating to interpretable metrics. The output gives the ability to glimpse higher dimensional spaces, which are normally too complex for humans to understand, and is a powerful tool in understanding the underlying features generating the metrics.

Acknowledgements

Thanks to Illumina® for supporting this research. The extremely high dimensional problems posed by genomic analysis provided the inspiration for this work.

2 Introduction

When training a model to make predictions in situations where interpretability is not required, the main goal is to maximise predictive accuracy [1]. Often black box models have a higher predictive accuracy than more interpretable solutions, particularly in higher dimensional or more complex problems [2]. Because of the uninterpretability of these models, they become impractical for situations where it is a necessity to understand why a model has chosen a particular output as its prediction, and what effect changes to values of the input features will have on the output of the model.

Some examples generic uses of model interpretation in real world applications include

1. Understanding the quality of a systems current understanding [3].
2. Interpreting how the value of a feature, or subset of features, affects a model's prediction (henceforth referred to as "feature interpretation"). For example, in a medical setting this interpretation can be used for suggesting the relation between features and a diagnosis. In another example, for manufacturing this interpretation can provide target values for the underlying processes involved in quality of produce.
3. The ability to use a model when not all values for the input features are present. By integrating across features which are not present, the certainty of a model's prediction can be estimated. Current methods, such as imputing values for features which are not present, ignores the variability of the absent features across the feature space, which LDR can provide.

3 The Algorithm

1. Scale the original data so it falls into [0, 1] intervals.
2. Train a predictive model on the scaled data.
3. *(Optional)* [1] Train an outlier detection predictive model on the scaled data.
4. Create a KDE (kernel density estimate) [4] of the training samples with a bandwidth h (In most cases this is best set equal to $\frac{1}{r}$). *(Optional)* to improve the visualization, normalize the training to produce a constant density output.
5. Sample n new points from the kernel density, using the predictive model to make a prediction at each point. *(Optional)* Weight the prediction according to the outlier classifier, where outliers have no model certainty in the prediction at that point.
6. Bin the samples according to regular intervals. For each dimension, group points with resolution r, reducing the value of the bin to the mean prediction across it.

[1] *(Optional)* here indicates that this is a sensible step when using typical predictive models. These steps are however not necessary as LDR can be used to estimate reduced dimensionality across any space.

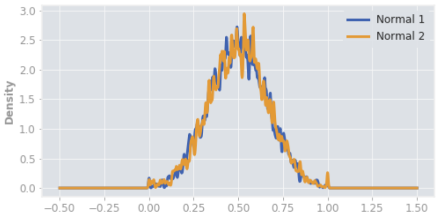

Figure 1: KDE (bandwidth $\frac{1}{r=50}$) of generated normal distributions, $\mu = 0.5$, $\sigma = \frac{1}{6}$.

4 Generated Data Classification Example

5000 samples are generated for two features from a 2 dimensional normal distribution with a mean of 0.5 and standard distribution of $\frac{1}{6}$, with any values outside the unit interval being scaled to the closest value in the interval. The KDE of each feature can be seen in figure 1. Classes are assigned by splitting the *Normal 1* feature in two, with one class falling to the left side of the peak and the other to the right. The data is then split across a 0.7/0.3 training/testing split. A single DT (decision tree) achieved an unsurprising accuracy of 100%.

Applying LDR with a resolution of 50 to the raw decision tree prediction results in the model interpretation shown in figure 2. An IF (Isolation Forest) [5] with a dynamically learned frontier was trained for outlier detection, which resulted in the outlier interpretation shown in figure 3. Interpolationg the DT and IF, as defined in 7.2, resulted in in the interpretation shown in figure 4. This shows how the *Normal 1* feature by itself can be a strong indicator of model prediction, while the *Normal 2* feature has little effect on the prediction by itself. Inspecting the predictions across the entire feature space gives the scatter plot shown in figure 5, showing that LDR is not really necessary for interpreting the model in this scenario; the next examples demonstrate how LDR fares on difficult to interpret models.

5 Classification Example

The Wisonsin breast cancer dataset [6] (provided by Scikit [7]) is used as it has 31 dimensions, a significant number. The aim with the dataset is to classify the tumor as either malignant or benign. There are 569 samples in total, with 357 benign and 212 malignant.

5.1 Random Forest and Isolation Forest Interpolation

A 100 estimator RF (Random Forest) [8] achieved an F1 score of 0.975, while an IF with a dynamically learned frontier was trained for outlier detection. Benign was selected as the positive category, resulting in a value of 1.0 indicating benign tumors and 0.0 inidicating malignant. All of the data is min-max scaled, then a 0.7/0.3 training/testing data split is applied. LDR is applied to the training data, integrating over

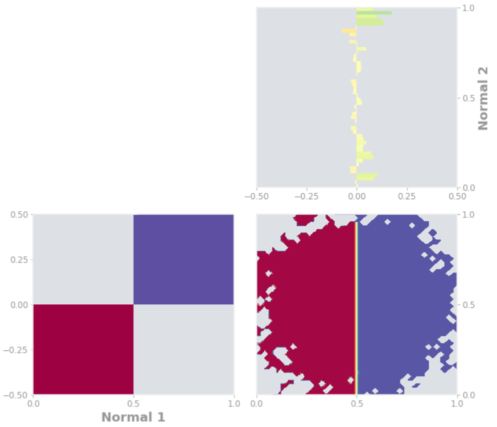

Figure 2: DT model prediction of generated data KDE samples. Red is certainty of class 1, blue certainty of class 2, yellow uncertainty, and grey a lack of samples in the KDE space. The top and left bar graphs are the single dimensional decomposition, while the bottom right is the two dimensional decomposition. In the single dimension estimates, the resulting value is shifted down by 0.5 to create a flat midpoint for the bars. The axis are labelled with the relative input minumum and maximum values used for scaling.

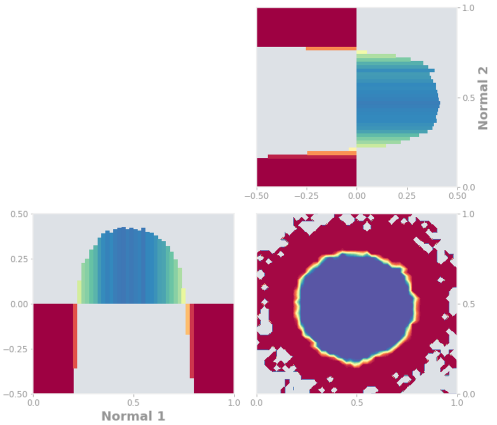

Figure 3: IF model prediction of generated data KDE samples. Red is certainty of outliers, and blue certainty of not outliers.

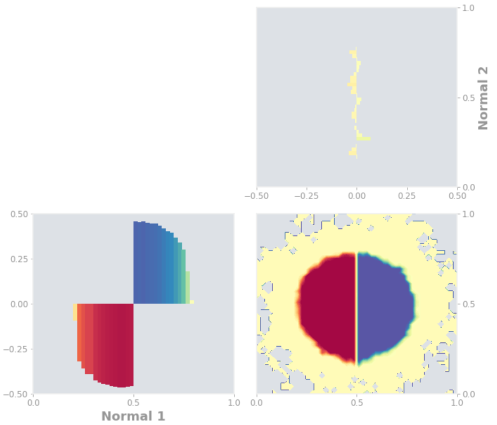

Figure 4: DT and IF model interpolation prediction of generated data KDE samples.

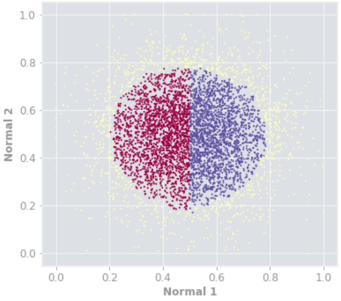

Figure 5: Naive scatter plot of model predictions on raw generated data.

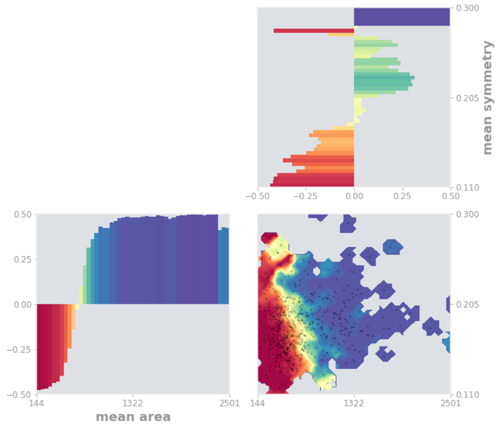

Figure 6: RF prediction certainty, where red indicates certaincy of malignancy, and blue certainty of benality.

50,000 randomly selected points from the weighted kernel density of the sample space with a resolution of 50. Mean symmetry and mean area are selected for visual inspection of their individual effect on the model's classification. The certainty of the RF is shown in figure 6, while the certianty of the IF is shown in figure 7. The final combined certainty is shown in figure 8.

5.2 Neural Network and IF Interpolation

A 3 layer feed forward Neural Network (NN) was used (via PyTorch [9]), consisting of an input layer, a hidden layer where tanh is applied, and an output layer. Softmax is applied when making predictions, and the final certainty of the prediction is calculated by calculating the proportion of the largest weighted class out of all weightings. The NN was trained over 50,000 epochs and achieved an F1 score of 0.961, slightly worse than the RF. The resulting LDR of the same features can be seen in figure 9

5.3 Analysis of Discrepancy Between RF and NN

There is very little discrepancy between the two LDR representations, both of which have similar predictive success. This is significant because random forests are widely regarded as more interpretable than neural net-

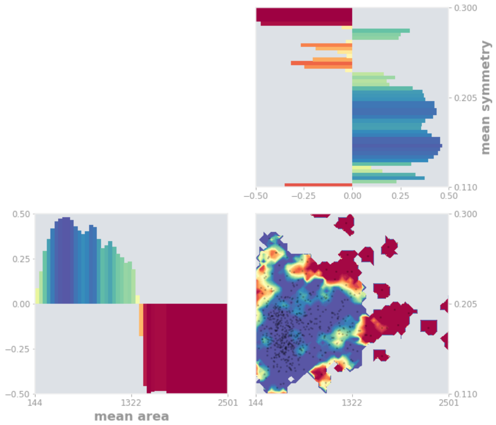

Figure 7: IF prediction certainty, where red indicates certianty of outlier, blue certainty of not outlier, and yellow uncertainty.

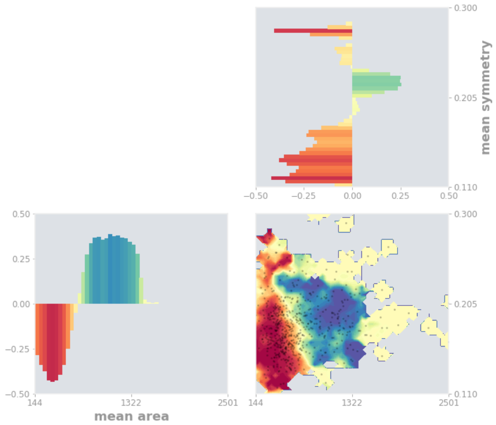

Figure 8: RF with IF interpolation prediction certainty, where red indicates overall certainty of malignancy, blue indicates overall certainty of benality, and yellow general overall uncertainty.

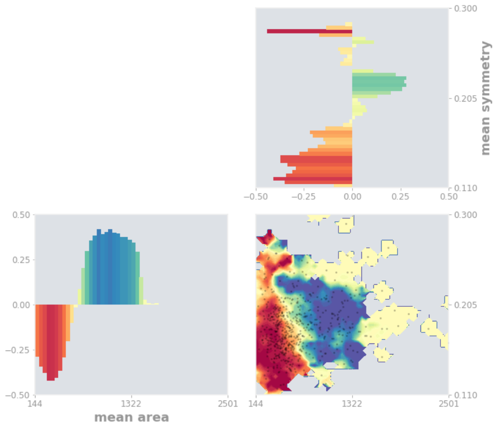

Figure 9: NN with IF interpolation prediction certanty.

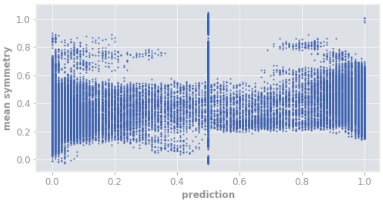

Figure 10: RF prediction certainty scatter plot of mean symmetry.

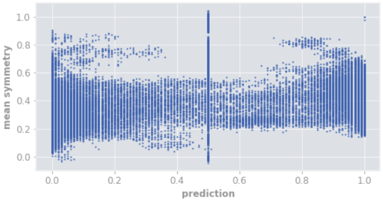

Figure 11: NN prediction certainty scatter plot of mean symmetry.

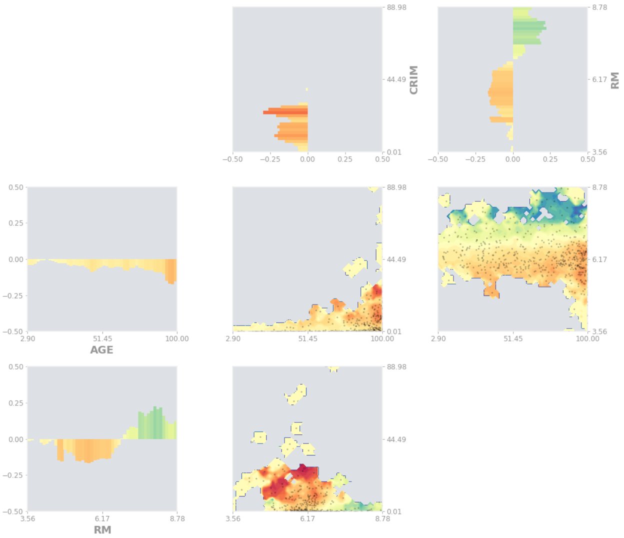

Figure 12: Regression RF with IF interpolation. AGE is the age of the house, RM the number of rooms, and CRIM the per capita crime rate per town.

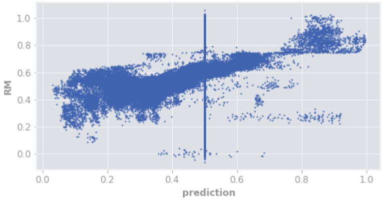

Figure 13: RF prediction certainty of number of rooms.

works [10], but LDR represents them both in an equally interpretable fashion. The only visual difference between the two LDR representation arises from the NN tending to be more certain of its classifications. This can be seen in the scatter plots of prediction certainties for mean symmetry of the RF in figure 10m compared to the NN in figure 11.

6 Regression Example

A similar method as with the classification was applied to the Boston House Price dataset [11] (also provided by Scikit). This dataset has 14 dimensions, where the target is to predict the house price based off of quantitative factors such as age of the house, the number of rooms, and the per capita crime rate of the town within which it is situated. Some interesting relations are made immediately available from the LDR, shown in figure 12, are

1. An increasing crime rate tends to correlated with houses being worth less. Above the median level of crime however, the model becomes uncertain.
2. Any quantity of crime only has a negative effect on house price.
3. The lower the crime rate, the more rooms a house tends to have, and the more it is worth.
4. The older the house, the less it tends to be worth.

The prediction certainty prior to be binning can be seen in figure 13.

7 Mathematical Explanation

7.1 Feature Space and Subsets Definition

Let a scaled feature space Ω for samples $w = \{0, 1, \dots, l\}$ be described as

$$\Omega = \begin{bmatrix} \omega_1 = \{d_{1,1}, d_{2,1}, \dots, d_{l,1}\} \\ \omega_2 = \{d_{1,2}, d_{2,2}, \dots, d_{l,2}\} \\ \vdots \\ \omega_m = \{d_{1,m}, d_{2,m}, \dots, d_{l,m}\} \end{bmatrix}$$

where $d_{i,j} \in [0, 1]$, l is the number of features and m is the number of samples in the feature space.

The subset X of values for features $t \subset w$ which are being used to predict a set Y of features $l - t = \{i, (i \in w) \wedge (i \notin t)\}$ are therefore defined as

$$X = \begin{bmatrix} x_1 = \{d_{1,1}, d_{2,1}, \dots, d_{|t|,1}\} \\ x_2 = \{d_{1,2}, d_{2,2}, \dots, d_{|t|,2}\} \\ \vdots \\ x_m = \{d_{1,m}, d_{2,m}, \dots, d_{|t|,m}\} \end{bmatrix}$$

$$Y = \begin{bmatrix} y_1 = \{d_{1,1}, d_{2,1}, \dots, d_{|l-t|,1}\} \\ y_2 = \{d_{1,2}, d_{2,2}, \dots, d_{|l-t|,2}\} \\ \vdots \\ y_m = \{d_{1,m}, d_{2,m}, \dots, d_{|l-t|,m}\} \end{bmatrix}$$

7.2 (Optional) Model and Outlier Prediction Interpolation

Let p describe a model prediction mapping, such that $p(x_i) \approx y_i$. Let o describe an outlier prediction mapping such that $o(x_i) = [1, 0]$, where 0 indicates complete certainty of x_i being an outlier and 1 complete certainty of x_i not being an outlier. Let outlier weighting and model prediction interpolation function f be defined as

$$f(x_i) = (p(x_i) - 0.5) \times o(x_i) + 0.5.$$

7.3 KDE Sample Drawing

The KDE with bandwidth $1/r$ can be estimated and n samples drawn from it in unison by drawing from the individual weighting each sample $x \in X$ contributes to the kernel density. This is done by randomly selecting n samples from X and drawing distributed points $q \sim N(x, 1/r)$. This sampling is equivalent to the VEGAS algorithm for the subsequent monte carlo integration [12]. The set of samples Q can therefore be described as

$$Q = \{q_1, q_2, \ldots, q_n\}.$$

Note that the VEGAS algorithm relies on the assumption that the sample space is univariate and i.i.d, resulting in LDR only successfully interpreting samples from the same feature space.

7.4 Monte Carlo Integration Over Feature Subsets

Let e be the set of intervals describing a single dimensional set of bins, such that

$$e = \{[0, \frac{1}{r}], (\frac{1}{r}, \frac{2}{r}], \ldots, (\frac{r-1}{r}, 1]\}.$$

Let $b \subset t$ describe the feature subspace being interpreted. Let E describe the $|b|$ multivariate permutations of bin pairs, such that

$$E = perm(e_1, e_2, \ldots, e_{|b|}).$$

The Monte Carlo integration of a single point of the feature subset can therefore be described, where Q_k represents all elements that fall within the bin E_k, as

$$I_{E_k} = \frac{1}{z} \sum_{i=1}^{z} f(q_i), q_i \in Q_k.$$

The entire feature subset can be interpreted by applying this for all $E_K \in k$.

8 Conclusion

I have demonstrated that it is possible to interpret a models understanding of high dimensional features spaces by estimating the certainty of the model over lower dimensional subsets of the feature space. However, before acceptance for clinical purposes, the model should be rigorously tested on more data sets. In addition to this, further research of extending LDR could be to interpret latent features that do not relate to uninterpretable input features, such as neural networks trained on images; a visualization of the prediction of the model based on a single pixel would not be of much use by itself.

References

[1] Leo Breiman et al. “Statistical modeling: The two cultures (with comments and a rejoinder by the author)”. In: *Statistical science* 16.3 (2001), pp. 199–231.

[2] Jürgen Schmidhuber. “Deep learning in neural networks: An overview”. In: *Neural networks* 61 (2015), pp. 85–117.

[3] Saleema Amershi et al. “Effective end-user interaction with machine learning”. In: *Twenty-Fifth AAAI Conference on Artificial Intelligence*. 2011.

[4] Emanuel Parzen. “On estimation of a probability density function and mode”. In: *The annals of mathematical statistics* 33.3 (1962), pp. 1065–1076.

[5] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. “Isolation forest”. In: *2008 Eighth IEEE International Conference on Data Mining*. IEEE. 2008, pp. 413–422.

[6] Olvi L. Mangasarian Dr. William H. Wolberg W. Nick Street. *Breast Cancer Wisconsin (Diagnostic) Data Set*. Nov. 1995. DOI: `https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)`.

[7] Fabian Pedregosa et al. “Scikit-learn: Machine learning in Python”. In: *Journal of machine learning research* 12.Oct (2011), pp. 2825–2830.

[8] Andy Liaw, Matthew Wiener, et al. “Classification and regression by randomForest”. In: *R news* 2.3 (2002), pp. 18–22.

[9] Adam Paszke et al. “Automatic Differentiation in PyTorch”. In: *NIPS Autodiff Workshop*. 2017.

[10] Lin Song, Peter Langfelder, and Steve Horvath. “Random generalized linear model: a highly accurate and interpretable ensemble predictor”. In: *BMC bioinformatics* 14.1 (2013), p. 5.

[11] D. Harrison and D.L. Rubinfield. *The Boston house-price data*. Nov. 1978. DOI: `http://lib.stat.cmu.edu/datasets/boston`.

[12] G Peter Lepage. “A new algorithm for adaptive multidimensional integration”. In: *Journal of Computational Physics* 27.2 (1978), pp. 192–203.