concept activation vector (CAV)

Authors

Alexander Jung*, Department of Computer Science, Aalto University, Espoo, Finland
*Corresponding author: alex.jung@aalto.fi

Abstract

Definition

A deep net labels photographs, and one of its labels is zebra. A user wants to know whether it arrives at that label through stripes. Stripes are not a feature the network was built with, and no single neuron need stand for them, so the question cannot be answered by reading off one number. It can be answered by supplying examples: photographs that show stripes, and photographs that do not (Kim et al., 2018, Sect. 3.1).

Consider such a deep net, consisting of several hidden layers, which is trained to predict the label of a data point from its feature vector. One way to explain the behavior of the trained deep net is by using the activations of a hidden layer as a new feature vector $\mathbf{z}$. The geometry of the resulting new feature space is then probed by applying the deep net to data points that represent a specific concept $\mathcal{C}$. By applying the deep net also to data points that do not belong to this concept, a binary linear classifier can be trained $g(\mathbf{z})$ that distinguishes between concept and non-concept data points based on the activations of the hidden layer. The resulting decision boundary is a hyperplane whose normal vector is the CAV (Kim et al., 2018, Sect. 3.2) for the concept $\mathcal{C}$. In the terminology of mechanistic interpretability, this direction is a feature of the trained deep net: a human-interpretable concept encoded as a direction in the space of activations.

Fig. 1 draws both objects in one plane, the one spanned by the activations of two neurons of a hidden layer. The decision boundary of the deep net curves through that plane; a straight line fitted to the same labels classifies only 82% of them correctly, which is what makes the boundary genuinely nonlinear. The concept, by contrast, occupies a half-plane: a linear classifier separates concept from non-concept activations with accuracy 0.96, and its weight vector is the CAV.

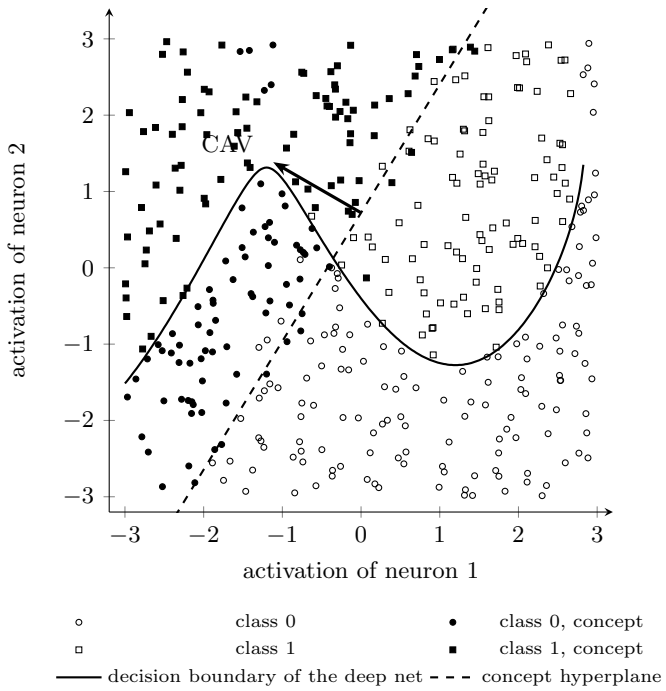

Figure 1: A concept is a direction, the class boundary is a curve. Both are drawn in the plane spanned by the activations of two neurons of a hidden layer. Circles and squares mark the two classes the deep net separates, filled markers the data points carrying the concept. The dashed hyperplane is the decision boundary of a linear classifier fitted to concept against non-concept activations; the CAV is its normal vector. Data generated by `pythondemos/cav.py`

A CAV is a global explanation: one direction for the whole hypothesis, not an account of a single prediction (see explanation). Like any explanation it has to be understandable and faithful. Understandable it is by construction, since the user supplies the examples that define the concept and can add more. Faithful is a separate question, and it is what testing with CAVs answers: the sensitivity of a prediction to the concept is the directional derivative of the score along the CAV (Kim et al., 2018, Sect. 3.3), so a concept the network does not use scores near chance. In the plane of Fig. 1, moving along the CAV raises the score at 65% of the class-1 data points, against 52% for a concept planted orthogonal to the direction the network varies in and 49% for random directions.

Cross References

deep net, feature, linear classifier, trustworthy AI, interpretability, mechanistic interpretability, transparency, explanation, explainability, neuron, decision

boundary, hyperplane

References

1. Kim et al. (2018). Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). In: Proc. 35th Int. Conf. Mach. Learn., pp. 2668–2677.